

国际机械工程先进技术译丛



移动机器人学科学方法

——智能体行为的量化分析

Scientific Methods in Mobile Robotics





（德）乌尔里希·内姆佐夫（Ulrich Nehmzow） 著

张文增 译







机械工业出版社
CHINA MACHINE PRESS

国际机械工程先进技术译丛

移动机器人学科学方法

——智能体行为的量化分析

(德)乌尔里希·内姆佐夫（Ulrich Nehmzow）著

张文增 译



机 械 工 业 出 版 社

本书原著是由自主移动机器人领域知名人士 Ulrich Nehmzow 博士撰写，在英国很受欢迎，并且在意大利、西班牙和加拿大等国被多所高等院校作为教材使用。

与移动机器人传统处理方式不同，本书基于动力系统、混沌、神经网络、统计学和系统辨识等理论，第一次运用量化分析和计算机建模对机器人、任务和环境三者交互行为建立了完整统一的形式化理论研究体系。这是一个全新的前沿视角，也是解决此类长期困扰我们的复杂问题的全新途径。

本书作为反映最新前沿技术的书籍，从基本概念到应用实例进行了系统的阐述，内容全面、重点突出、理论与实例并重，叙述风格深入浅出，能够很好地将理论介绍给相关领域的一般研究人员和专家学者。

图书在版编目（CIP）数据

移动机器人学科学方法：智能体行为的量化分析/（德）内姆佐夫（Nehmzow，U.）著；张文增译.—北京：机械工业出版社，2009.11
（国际机械工程先进技术译丛）
ISBN 978-7-111-28976-0

Ⅰ.移… Ⅱ.①内…②张… Ⅲ.移动式机器人-研究 Ⅳ.TP242

中国版本图书馆 CIP 数据核字（2009）第 203346 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）
策划编辑：孔 劲 责任编辑：舒 雯 版式设计：霍永明
封面设计：鞠 杨 责任校对：张 媛 责任印制：乔 宇
北京机工印刷厂印刷（兴文装订厂装订）
2010 年 1 月第 1 版第 1 次印刷
169mm×239mm · 12.5 印张 · 240 千字
0 001—3 000 册
标准书号：ISBN 978-7-111-28976-0
定价：38.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务	网络服务
社服务中心：（010）88361066	门户网：http://www.cmpbook.com
销 售 一 部：（010）68326294	
销 售 二 部：（010）88379649	教材网：http://www.cmpedu.com
读者服务部：（010）68993821	**封面无防伪标均为盗版**

献给 RobotMODIC 项目组：

史蒂夫·比灵斯（Steve Billings）、司奥查理士·凯莉奥考（Theocharis Kyriacou）、罗伯特·伊格莱西亚斯·罗德里格（Roberto Iglesias Rodriguez）、凯斯·沃克（Keith Walker）和胡戈·维埃拉·内图（Hugo Vieira Neto）、以及她的支持团队：

克劳迪娅（Claudia）和亨里雅塔·内姆佐夫（Henrietta Nehmzow）、玛利亚·凯瑞考（Maria Kyriacou）、米凯莱·维埃拉（Michele Vieira）和玛克辛·沃克（Maxine Walker）

译丛序言

一、制造技术长盛永恒

先进制造技术是20世纪80年代提出的，由机械制造技术发展而来。通常可以认为它是将机械、电子、信息、材料、能源和管理等方面的技术，进行交叉、融合和集成，综合应用于产品全生命周期的制造全过程，包括市场需求、产品设计、工艺设计、加工装配、检测、销售、使用、维修、报废处理、回收利用等，以实现优质、敏捷、高效、低耗、清洁生产，快速响应市场的需求。因此，当前的先进制造技术是以产品为中心，以光机电一体化的机械制造技术为主体，以广义制造为手段，具有先进性和时代感。

制造技术是一个永恒的主题，与社会发展密切相关，是设想、概念、科学技术物化的基础和手段，是所有工业的支柱，是国家经济与国防实力的体现，是国家工业化的关键。现代制造技术是当前世界各国研究和发展的主题，特别是在市场经济高度发展的今天，它更占有十分重要的地位。

信息技术的发展并引入到制造技术，使制造技术产生了革命性的变化，出现了制造系统和制造科学。制造系统由物质流、能量流和信息流组成，物质流是本质，能量流是动力，信息流是控制；制造技术与系统论、方法论、信息论、控制论和协同论相结合就形成了新的制造学科。

制造技术的覆盖面极广，涉及到机械、电子、计算机、冶金、建筑、水利、电子、运载、农业以及化学、物理学、材料学、管理科学等领域。各个行业都需要制造业的支持，制造技术既有普遍性、基础性的一面，又有特殊性、专业性的一面，制造技术具有共性，又有个性。

我国的制造业涉及以下三方面的领域：

- 机械、电子制造业，包括机床、专用设备、交通运输工具、机械设备、电子通信设备、仪器等；
- 资源加工工业，包括石油化工、化学纤维、橡胶、塑料等；
- 轻纺工业，包括服装、纺织、皮革、印刷等。

目前世界先进制造技术沿着全球化、绿色化、高技术化、信息化、个性化和服务化、集群化六个方面发展，在加工技术上主要有超精密加工技术、纳米加工技术、数控加工技术、极限加工技术、绿色加工技术等，在制造模式上主要有自动化、集成化、柔性化、敏捷化、虚拟化、网络化、智能化、协作化和绿色化等。

二、图书交流源远流长

近年来，国际间的交流与合作对制造业领域的发展、技术进步及重大关键技术的突破起到了积极的促进作用，制造业科技人员需要及时了解国外相关技术领域的最新发展状况、成果取得情况及先进技术应用情况等。

必须看到，我国制造业与工业发达国家相比，仍存在较大差距。因此必须加强原始创新，在实践中继承和改造，学习国外的先进制造技术和经验，提高自主创新能力，形成自己的创新体系。

国家、地区间的学术、技术交流已有很长的历史，可以追溯到唐朝甚至更远一些，唐玄奘去印度取经可以说是一段典型的图书交流佳话。图书资料是一种传统、永恒、有效的学术、技术交流方式，早在20世纪初期，我国清代学者严复就翻译了英国学者赫胥黎所著的《天演论》，其后学者周建人翻译了英国学者达尔文所著的《物种起源》，对我国自然科学的发展起到了很大的推动作用。

图书是一种信息载体，图书是一个海洋，虽然现在已有网络、光盘、计算机等信息传输和储存手段，但图书更具有广泛性、适应性、系统性、持久性和经济性，看书总比在计算机上看资料要方便，不同层次的要求可以参考不同层次的图书，不同职业的人员可以参考不同类型的技术图书，同时它具有比较长期的参考价值和收藏价值。当然，技术图书的交流具有时间上的滞后性，不够及时，翻译的质量也是个关键问题，需要及时、快速、高质量的出版工作支持。

机械工业出版社希望能够在先进制造技术的引进、消化、吸收、创新方面为广大读者作出贡献，为我国的制造业科技人员引进、纳新国外先进制造技术的出版资源，翻译出版国际上优秀的制造业先进技术著作，从而能够提升我国制造业的自主创新能力，引导和推进科研与实践水平不断进步。

三、选译严谨质高面广

1）精品重点高质　本套丛书作为我社的精品重点书，在内容、编辑、装帧设计等方面追求高质量，力求为读者奉献一套高品质的丛书。

2）专家选译把关　本套丛书的选书、翻译工作均由国内相关专业的专家、教授、工程技术人员承担，充分保证了内容的先进性、适用性和翻译质量。

3）引纳地区广泛　主要从制造业比较发达的国家引进一系列先进制造技术图书，组成一套“国际机械工程先进技术译丛”。当然其他国家的优秀制造科技图书也在选择之内。

4）内容先进丰富　在内容上应具有先进性、经典性、广泛性，应能代表相关专业的技术前沿，对生产实践有较强的指导、借鉴作用。本套丛书尽量涵盖制造业各行业，例如机械、材料、能源等，既包括对传统技术的改进，又包括新的设计方法、制造工艺等技术。

5）读者层次面广　面对的读者对象主要是制造业企业、科研院所的专家、

研究人员和工程技术人员，高等院校的教师和学生，可以按照不同层次和水平要求各取所需。

四、衷心感谢不吝指教

首先要感谢许多积极热心支持出版“国际机械工程先进技术译丛”的专家学者，积极推荐国外相关优秀图书，仔细评审外文原版书，推荐评审和翻译的知名专家，特别要感谢承担翻译工作的译者，对各位专家学者所付出的辛勤劳动表示深切敬意，同时要感谢国外各家出版社版权工作人员的热心支持。

本套丛书希望能对广大读者的工作提供切实的帮助，欢迎广大读者不吝指教，提出宝贵意见和建议。

机械工业出版社

原 书 序

移动机器人广泛地应用于运输、监视和医疗等一系列领域。在所有这些应用中，分析和控制移动机器人的性能显然是非常重要的，而让人惊讶的是目前仍难以找到实现这一目标的公式化方法。本书介绍了基于统计学、动力系统理论和系统辨识的，可以解决这些问题的方法和过程。核心目标是力图以一种易懂的方式解释清楚机器人、任务和环境之间的交互性，以便于以一种系统化和结构化的方式来分析系统特征、设计控制器以及复制行为。实现任务的移动机器人构建的公式化方法代表着一种解决此类复杂问题的全新途径。

内姆佐夫（Nehmzow）博士已经杰出地完成了建立和描述一个统一框架的工作，该框架清楚地陈述了建立移动机器人理论的关键问题。得益于精心的题目组织和清楚的说明，本书很好地介绍了此专业领域的一些新方向。与移动机器人传统处理方式不同，内姆佐夫博士对一些长期存在的问题提供了一种全新的视角。我确信此专业领域中激动人心的崭新一页才刚刚开始。本书所涉及的概念都十分易懂，它一定会带来广泛的反响，我相信未来若干年这都会是一本非常有价值的参考书。

史蒂芬 A. 比灵斯（S. A. Billings）

译 者 序

机器人学是一门交叉学科，涉及机械、控制、传感器技术、信号处理、模式识别、人工智能和计算机技术等学科。近年来，移动机器人特别是自主式移动机器人成为机器人研究领域的重点之一。自主式移动机器人，即能够按照预先给定的任务指令，根据已知的地图信息作出全局的路径规划，并在行进过程中不断感知周围的局部环境信息，自主地作出各种决策，引导自身安全运动，并执行要求的动作和操作。自1961年美国Unimation公司研制出世界上第一台往复式工业机器人以来，机器人的发展经历了三个阶段：第一代示教再现机器人，第二代传感控制机器人，第三代智能机器人，其中第三代智能机器人是人类的最高梦想，也最受人关注。

美国麻省理工学院（MIT）人工智能研究所的创始人明斯基教授（M. Minsky）长期担任所长一职，是人工智能研究的象征性人物，在智能机器人、人工智能、计算机领域享有盛誉，曾获得图灵奖。他最著名的成就是20世纪70年代初期发表的有关于智能的一般说明性的理论框架。他在《心智社会》（Society of Mind）这本书中最早提出智能体（Agent）这一概念，用它来描述一个具有自适应、自治能力的硬件、软件或其他任何自然物和人造物，从而在人工智能各领域中掀起了研究基于智能体（Agent-based）的技术和基于智能体的系统（Agent Architecture）的高潮，并逐渐为心理学、生物学、物理学等学科所应用。

20世纪80年代初，美国麻省理工学院（MIT）布鲁克斯教授（R. A. Brooks）更是将智能机器人和人工智能的研究推向了一个广泛研究的高潮，他们特别提出了现场人工智能（Situated AI）的全新概念。他们认为，传统的孤立地研究智能现象的方法具有很大局限性。他们主张应该从智能体与环境的交互中（即现场中）研究智能现象，以解决人工智能中的无数困难问题。这种新的人工智能研究思路，有人把它称为“自主智能体研究”（Autonomous Agent Research），也有人称之为“基于行为的人工智能”或“自下而上的人工智能”，以区别于经典的“基于知识的人工智能”，或“自上而下的人工智能”。

21世纪，计算机网络为人工智能带来了第二个黄金时代。计算机网络一方面丰富了人工智能的某些研究领域，另一方面也提出了许多新的人工智能领域非常重要的研究课题——基于智能体的技术（Agent-based Technology）就是其中非常热门的研究前沿课题。

本书作者长期致力于智能体行为量化分析研究，受邀在多个国家的许多大学讲授本书的主要内容，受到普遍欢迎。本书的最大的成就在于基于统计学和现代的动力系统、混沌、神经网络及系统辨识等理论，第一次运用量化分析和计算机建模建立了对移动机器人、任务和环境的交互性的完整、统一的形式化理论框架体系，并给出了大量移动机器人学实例以及对它们的量化分析，从而成为解决移动机器人和智能体复杂行为问题的重要途径。同时本书从基本概念到应用实例进行了系统的阐述，内容全面、重点突出、理论与实例并重，叙述风格深入浅出，能够很好将理论介绍给相关领域的一般研究人员和专家学者，可作为高等院校机器人、自动控制、计算机、人工智能、模式识别和生物医学工程等专业的机器人课程或人机交互行为课程的教材，也可作为从事机器人相关工作的科研及技术人员的参考书。

张文增
于清华园

前　　言

本书是关于行为研究的科学方法，这里的“行为”代表任何“有行为”智能体的行为，它可以是生命，也可以是机器。因此，它既包括了机器人行为的分析，又不严格局限于此。本书中所讨论的内容同时适合于生物学家和机器人学家。

这里的“科学方法”代表着获得系统知识的原理和过程［Merriam Webster, 2005］，包括以下一些方面：

1）问题的辨识与设定；

2）实验过程，包括实验的设计、观察的过程、数据的采集和解释；

3）假设的设定与检验。

本书所提出的假设是行为（主要指运动）可以被量化地描述和分析，并且这些量化的描述可以用于支持原理性的研究、实验的重复和独立验证。

本书本身就是一个实验。除了分析智能体行为外，作为机器人学领域的实践者，我们还将机器人学研究的实践扩展到行为的精确描述，并对它进行可检验的预测，再到实验结果的独立复制和验证的一整套标准过程中。

我非常高兴介绍本书的内容。它们开辟了机器人学研究的一条新路，这条路将把读者引入生动活泼、激动人心和富有成果的讨论与新研究中（6.7 节中的 Java 机器人就是其中一例）。用量化的方法解释实验结果的研究方式使我们实验室中产生了许多全新的实验方法。例如，不是简单地开发一个自我充电的机器人，而是力图找到一个基线，用它来作为比较实验结果的“标准”。这意味着，发表的论文将不再仅仅包含对于特殊结果的描述（存在性证明），还要与机器人学领域建立的基线进行量化比较。

至今对于这些争议的反响确实令人吃惊！似乎基本上不存在中间地带。关于在机器人学中采用科学方法的观点将研究者们划分为两个泾渭分明的阵营。我们来看看全面的反响：一方面，一个著名的机器人学方面的期刊甚至拒绝对任务辨识的相关论文进行同行评议，并且在还没有进行评审的情况下就已经拒绝发表了。另外，在一次研讨会上，听众差不多睡着了！另一方面，两天后同样的话题却引起了要求追加一个晚上以便第二天进一步讨论的结果（并且这是在两个小时的讨论之后提出的）。意大利巴勒莫大学（Palermo University）、西班牙圣地亚哥联合大学（Santiago de Compostela University）和加拿大纽芬兰纪念大学（Memorial Universtity Newfoundland）要求“机器人学科学方法”作为一个校

外课程，这改变了他们所有机器人学专业学生的时间表，并且学校还要对他们进行关于这些客座讲座的考核。

对于这个反响，我很受鼓舞，因为它们表明移动机器人学科学方法这一主题并不是盲目和随意提出的，也不是偏离机器人主题的，而是对于我们学科的一个重要延伸。本书的目的是找出并且鼓励有关该主题的理论化和系统化的科学讨论。如果你愿意享受一场很好的辩论，我相信你将会很喜欢本书！

致　谢

科学从来不会在孤立中完成，而是主要依靠外部协作，“铁磨铁，磨出刃来，朋友相感，也是如此”（《旧·箴》，27，17）。本书的诞生也证明了这一点。本书所列的实验和结果都是全世界同仁们合作的结果。他们中的许多人也因此成了朋友，我感激我所获得的所有支持和帮助。

本书所讨论的绝大多数实验是在英国埃塞克斯大学（Universtity of Essex）开展的，这儿有我们新机器人学研究实验室，该实验室提供了用于进行本书所作研究的优良设备。我从与埃塞克斯大学分析与辨识机器人学小组的每个人的讨论中受益匪浅，他们是：西奥·凯瑞考（Theo Kyriacou）、胡戈·维埃拉·内图（Hugo Vieira Neto）、里博·史贝克（Libor Spacek）、约翰·福特（John Ford）和顾东兵（Dongbing Gu）等，还有我的同事杰夫·雷诺兹（Jeff Reynolds）。本书的相当一部分实际上是写于访问澳大利亚卧龙岗大学（University of Wollongong）的飞利浦·迈凯洛（Phillip Mckerrow's）项目组期间，我感激他们的支持以及埃塞克斯大学的公休假。提到公休，美国圣地亚哥市洛马拿撒勒大学的凯斯·沃克（Keith Walker）和西班牙圣地亚哥联合大学电子和计算机科学系的罗伯特·伊格莱西亚斯·罗德里格斯（Roberto Iglesias Rodriguez）在他们到埃塞克斯大学的公休期间对本书作出了重要贡献。我还欠一些其他学科的同事的情，特别是生命科学学科的同事们，他们对本书所提出的方法在生物学、心理学等方面的应用给予的评论让我很感激。特别感谢德国法兰克福市歌德大学的沃尔夫冈（Wolfgang）和罗斯维萨·威尔兹克（Roswitha Wiltschko）以及他们团队的支持。

如果没有我的同事兼朋友——英国谢菲尔德大学（University of Sheffield）的史蒂芬·比灵斯（Steve Billings）的帮助和所承担的工作，我的同事兼朋友——西奥·凯瑞考（Theo Kyriacou）提供的帮助以及英国工程师与物理科学研究委员会的支持，将不会形成本书讨论工作的支柱——RobotMODIC（Robot Modelling, Identification and Characterisation）项目。我从所有的科学、技术、财务和道义上的支持中受益良多，感谢我的同事们和赞助者们。

最后，我要感谢所有我在德国的家人，感谢他们坚定、美好和慷慨的爱与支持。我的妻子克劳狄娅（Claudia），如支持我的第一本书一样，一路走来提供了许多建设性的帮助，而亨里雅塔（henrietta）乐衷于被“品评”。感谢你们！

乌尔里希·内姆佐夫（Ulrich Nehmzow）

目　录

第1章　移动机器人学简介

概要：本章简要介绍了移动机器人学，目的是为不熟悉此领域的读者提供必要的背景知识。

1.1　本书不是关于移动机器人学的

本书确实不是关于移动机器人学的！本书只是从移动机器人学的角度来写的，虽然所给的是移动机器人学的例子，但是它所解决的问题是关于行为分析的。此处，行为是一个非常宽泛的概念，它可能涉及到移动机器人的运动、机器人臂部的轨迹、在迷宫中打转的老鼠、飞回家的信鸽、高速公路上的交通或者数据网络中的通信。简而言之，本书关注动力系统行为的描述，它可以是物理实体，也可以是仿真的。它的目标是量化地分析行为、比较行为，建立模型并且作出预测。本书中的内容不仅与机器人学家有关，而且与心理学家、生物学家、工程师、物理学家和计算机科学家都有关。

鉴于本书所提到的例子都是来自移动机器人学领域，所以明智的做法是给阅读此书的所有非机器人学家做一个关于移动机器人的非常简短的介绍。有关移动机器人全面的讨论可以参看文献［Nehmzow，2003a］，如果此书被用作教材，建议学生首先阅读关于移动机器人学的总体介绍，参见文献［Nehmzow，2003a，Siegwart and Nourbakhsh，2004，Murphy，2000］。

1.2　什么是移动机器人学

图1.1是一个典型的移动机器人——麦哲伦专业版Radix，用于埃塞克斯大学的实验项目。该机器人配备有超过50个机载传感器和一台机载计算机，能够通过这些传感器感知环境，利用计算机处理信号，并且通过计算控制自身的空间运动。

Radix是完全自主的，这意味着它不依赖于外部世界的任何联系，它会自我充电，因此不需要电源的电缆线；它自身带有计算机，因此不需要外部控制机构的有线或无线的连接。它也无需人类的远程遥控，能自主地与它所处环境进行交互，并且无需外部干预而确定自己的运动。

不是所有的移动机器人都是自主的，但是所有的移动机器人都能够实现位

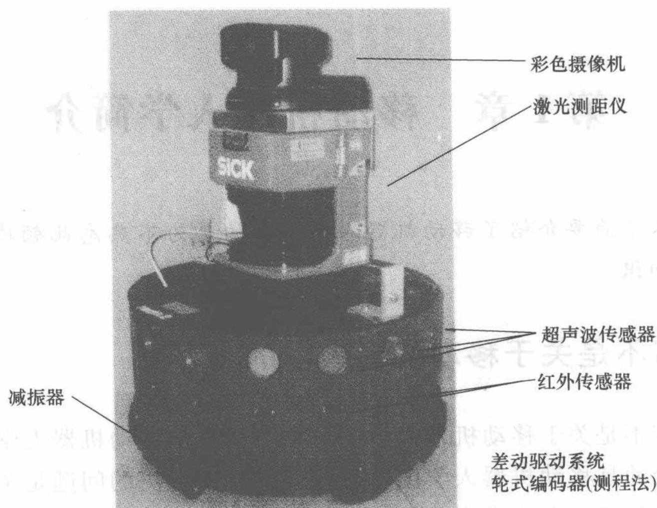


图 1.1 Radix（本书讨论的实验中使用的麦哲伦专业版移动机器人）

置移动。这可以通过使用腿或者轮子来实现，还有的移动机器人可以爬壁、游泳或者飞行。移动机器人学科关注的机器人控制问题有：它们如何能够实现被赋予的任务？如何将它们设计得更加可靠，能在存在传感器噪声、矛盾和错误信息的更大范围的环境条件中运行？这些正是移动机器人学要解决的问题。

1.2.1 工程

移动机器人由硬件构成：传感器、驱动器、电源、计算用的硬件、信号处理硬件、通信用的硬件等。这意味着在设计移动机器人时需要有很强的工程基础和关于机器人学工程方面的大量背景知识 [Critchlow, 1985, McKerrow, 1991, Fuller, 1999, Martin, 2001]。机器人学工程方面的期刊有很多，包括：《Advanced Robotics》、《Automation in Construction, Industrial Robot》、《IEEE Trans. on Robotics》、《IEEE Trans. On Automation Science and Engineering》、《International Journal of Robotics Research》、《Journal of Intelligent and Robotic Systems》、《Mechatronics》、《Robotica》、《Robotics and Autonomous Systems and Robotics》和《Computer Intergrated Manufacturing》等。

1.2.2 科学

自主移动机器人在感知与行动之间形成了一个闭环，它能通过传感器感知环境，使用机载的计算机处理信息，然后通过运动对此进行响应。这一方式带来一些有趣的问题，例如，如何实现“智能”行为的问题；什么是实现任务行为的基础；通过什么机制能够实现类似“智能”的行为。其次，机器人与环境

之间的交互和动物与环境间的交互很相似，能否复制动物的行为到机器人上，使得机器人更加成功呢？能否利用机器人来弄明白控制动物行为的机制？

关于行为的问题传统上是心理学家、行为学家和生物学家研究的范畴，我们称之为“科学”，这不是硬件和软件设计（工程上）的问题（例如机器人自身的问题），而是将移动机器人作为工具来研究其他问题的问题。移动机器人的应用在不断增长，这方面存在大量的文献，范围从自主智能体的“抽象”讨论（[Braithenberg, 1987, Steels, 1995, Von Randow, 1997, Ritter et al, 2000]）到人工智能的应用和机器人学的辨识科学（[Kurz, 1994, Arkin, 1998, Murphy, 2000] [Dudek and Jenkin, 2000]）。“Adaptive Behavior”或“IEEE Transactions on Systems, Man, and Cybernetics”也发表与此主题相关的文章。

1.2.3　（商业化）应用

移动机器人基本的优点使它成为很多商业应用中有吸引力的选择，如运输、检验、监视、医疗 [Katevas, 2001]、远程搬运以及危险环境操作、娱乐机器人（人造宠物），甚至博物馆导游 [Burgard et al, 1998]。

像其他任何移动或者固定的机器人一样，移动机器人都能够连续而不知疲倦地在不利条件下运行。它们可以运行在核辐射中、极端温度下、烟雾中、极端气压下或其他危险情况下。由于它们能够不受干扰、一周 7 天、每天 24 小时不间断地工作，因此即使非常高的投资也可以在相对较短的时间内收回成本，同时机器人不知疲倦的工作能力还减少了出错的风险。

除了上述所有的机器人都共有的能力外，移动机器人还有特别的优点，就是能够确定它们自身的位置。它们能够为了即将到来的任务而获取优化的工作位置。如果需要的话，它们可以在操作时随时改变位置（这是相关联的，例如机器人进行大结构的装配工作）。由于它们能搬运负载，因此它们是极其灵活的：移动机器人只要配备一个机载的操作臂，就能够根据工作需要扩大工具加工的范围，并将工具移到加工现场。它们能够携带测量仪器并根据需要将这些仪器应用到指定的位置（例如在规定的精确位置测量温度、气压和湿度等），已经开发的用于空间探索的移动机器人就能实现这一功能 [Iagnemma and Dubowsky, 2004]。

此外，协同工作的多移动机器人系统能够实现那些单台机器人难以完成的任务，例如，需要在适当的位置抓住一个物体进行焊接、放置电缆线或管件等。协作的多机器人学是一个激动人心的研究领域。文献 [Beni and Wang, 1989, Ueyama et al, 1992, Kube and Zhang, 1992, Arkin and Hobbs, 1992, Mataric, 1994 and Parker, 1994] 是这一领域的研究实例。

移动机器人也有一些独有的缺点，影响了它们在工业上的应用。

首先，移动机器人能够确定自身位置的显著优点带来了精度下降的缺点。尽管机器人操作臂和移动机器人都受到传感器和驱动器噪声的影响，但是由于移动机器人底盘运动的不精确，导致移动机器人的位置难以像永久固定在某处的机器人操作臂那样精确定位。而且，任何驱动系统都会有一定的游隙，它会影响精度的理论极限。

其次，移动机器人有难以预测的特点，特别是当它们在自主运动时，即完全没有外部连接（例如电源或者控制）时，很难预测。以我们目前对于机器人—环境交互过程的所知来说，不可能从解析的角度确定极端条件下的稳定性界限和行为。本书的一个目标是建立一个机器人—环境交互的理论，该理论允许对机器人操作进行理论分析，例如考虑极端条件下的稳定性和行为。

第三，任何一台移动机器人的负载都是有限度的，这与机载电源和操作次数都有关系。当前能实现的最高能量密度是采用内燃机，但是它难以在很多场合中应用，例如室内环境。也可以选择电驱动方式，它是依赖外部电源的，此种方式由于限制了机器人的运动范围，而与移动机器人的移动性——这一与生俱来的优点相矛盾了。或者采用机载电池，不过目前电池还相当的重。然而随着技术进步，这一缺点将会越来越微不足道。

1.3　行为的产生

为何移动机器人会按照它被要求的那样运行？移动机器人以某种方式被编程并放置在某些环境下去执行该程序，为何它会准确地遵循它所走的路径，而不是别的路径？

移动机器人的行为——当机器人与它所处环境发生交互时，什么被观察到——并不是单独由机器人编程产生的结果，而是由三方面的基本要素构成的：

1）机器人上所运行的程序（“任务”）。

2）机器人的物理组成（它的传感器、电动机工作方式、电池供电等）。

3）环境本身（对于机器人的传感器，可视物体是怎样的，轮子的抓地力如何等）。

机器人的行为产生于这三方面基本要素的交互作用，如图 1.2 所示。

这一观点是比较容易解释的。当机器人控制程序发生改变时，可以看到机器人行为显而易见随之发生改变。如果环境改变会发生什么呢？比如把一个避障移动机器人扔到游泳池里！很显然，所谓的“避障”是指在这样或那样的特定环境中可以躲避障碍。最后，如果改变机器人传感器，例如拔掉一个传感器的插头，机器人的行为也会随之发生改变。当我们谈论机器人行为时，实际上是同时在谈论任务、机器人和环境。移动机器人学科学方法的目的就是分析和

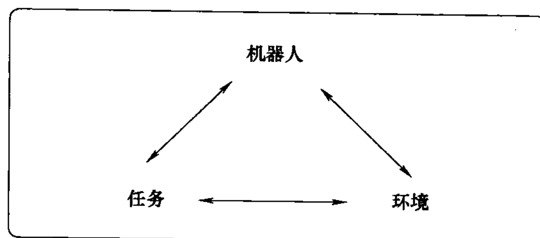


图 1.2　机器人—环境交互作用的基本三角形

理解产生行为的这三个基本因素之间的相互关系。

是什么造成机器人学很难?

移动机器人是一种嵌入式的、定位式的智能体。嵌入式是因为机器人通过其动作与它所在环境发生交互作用；定位式是因为它的动作影响着它未来所处的状态。与计算机仿真不同（甚至那些涉及伪随机数的仿真），由于传感器和驱

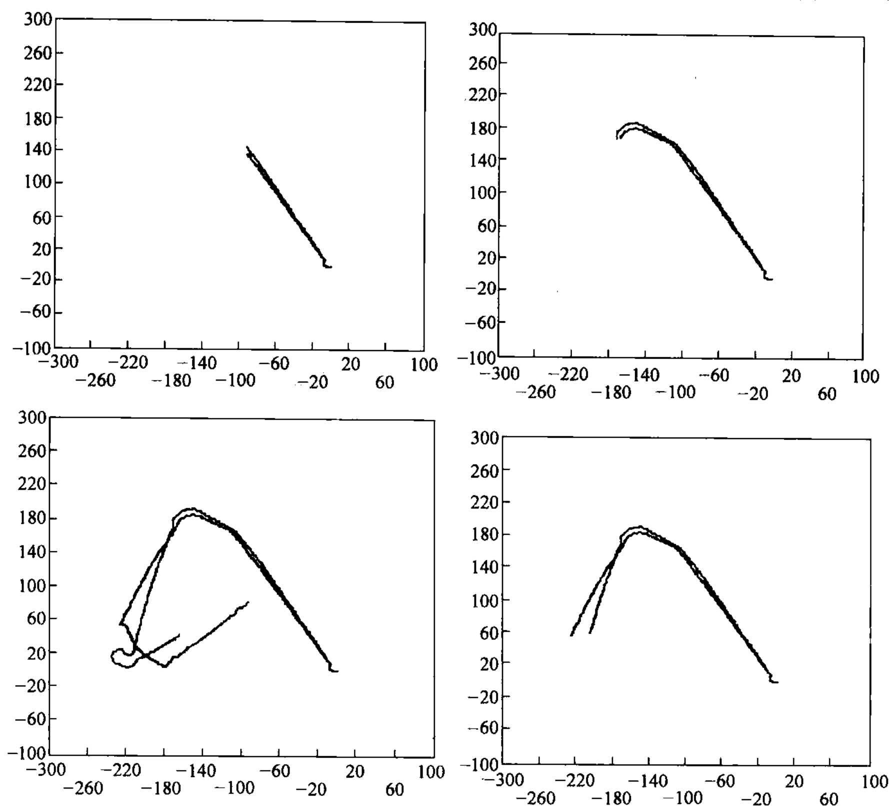


图 1.3　移动机器人的行为不总是能被预测（该图显示一段时间内的轨迹，从左上图开始沿顺时针顺序）

动器的噪声以及在许多动力系统中的混沌干扰，机器人与它的周遭环境之间的交互作用并不总能被预测。一个在真实世界中运行的移动机器人与一个计算机仿真模型的区别之处在于不可重复性。计算机仿真模型可以精确地重复，一遍又一遍，而对于移动机器人这是不可能的。

为了说明此现象，设计了一个简单的实验，图 1.3 是该实验的结果。一个移动机器人两次被放置在（尽可能的）同一个起点位置，在同样的环境中执行同一个程序。两次同样的实验在几分钟内分别进行。

如图 1.3 中所示，两次轨迹起点非常相似，但是在两三个拐弯后彼此就产生了明显的偏离。实验很短，这两个轨迹就已经非常不同，尽管实验设置一点没有改变。机器人没有改变，任务也没有改变，并且环境也没有改变。唯一不同的是两次的起点位置存在非常小的不同。

对于这两次令人惊讶的轨迹的不同解释是：很小的扰动（例如传感器噪声）很快加入进来，因为一个很小的不同感知将导致一个很小的不同的电动机响应，该响应又引起另一个不同的感知，如此循环影响下去，以至于不久就产生了不同的轨迹。正是这样的行为（可以称为“混沌的行为”，见第 4 章）使得“真实世界”机器人学很难建模，也正是这样的行为导致计算机仿真模型的预测与实际机器人的行为之间存在着差异。这不是机器人的错，而是机器人—环境交互的一个自然和正常的部分。行为不是智能体的性质，它是由智能体与其环境之间交互作用组成的一个动态过程 [Smithers，1995]。

图 1.4 是“真实世界”实验中观察到的现象，该实验是关于在某一段时间内机器人探索环境的。在机器人探索期间，机器人在不同时刻两次从指定的起点位置出发。最初，两个轨迹彼此很近，但是在第一个拐弯处可以观察到一个小的偏离（*A* 点）。在第二个拐弯处（*B* 点），偏离就放大了，并且在 *C* 点最初

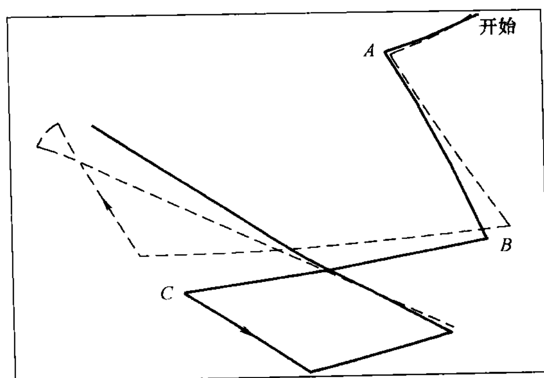


图 1.4 在“真实世界”实验中所观察到的两种轨迹，最初位置很接近，但几十秒钟后就产生了很大的偏离

接近的轨迹已改变为存在相当大的偏离，以至于机器人的动作从根本上发生了改变。图中实线显示的轨迹结果不可能被重复。

1.4　自主移动机器人学研究问题举例

本小节的目的在于通过举例的方式强调移动机器人应用的几个领域。本节仅给出一个移动机器人领域已经研究到什么程度的“感觉”，关于此主题的更详细的信息可以参见教材［Arkin，1998，Murphy，2000］和［Nehmzow，2003a］。

1.4.1　导航

如果没有导航的能力，那么移动性的优点不可能被完全挖掘出来。例如在生命王国里，人们很难找到一种能够移动但是却没有导航技能的动物。因此，导航是移动机器人学中一个重要的主题，颇受瞩目。

基于地图的导航存在以下所有或者部分性能［Nehmzow，2003a，Nehmzow，2003b］：

1）自我定位：如果不能标识出机器人在地图上的位置，任何导航都是不可能的。自我定位是所有导航的基础。

2）映射建立：此处“映射”这个词代表A和B空间的双向映射。A和B空间并不严格局限为导航映射，而是代表两个空间的任何一对一的映射（例如传感器的感知和人工神经网络的响应）。

3）映射解释：如果映射不可解释，那么该映射对于此智能体来说是无用的。因此映射解释与获取映射的能力同样重要。

4）路径规划：这一能力涉及到在一系列动作中作出决定，这些动作将使机器人从一处运动到另一处。路径规划通常至少包含了自我定位和映射解释。

5）校正：正如上面所述，机器人与真实世界的交互是不能被完全预测的，因而任何导航机器人都需要具备从误差中进行校正的能力。这通常涉及到要更新自我定位和路径规划，但是有时也采用特定的校正策略，比如回到一个已知的固定点，然后从那儿再重新开始导航。

移动机器人中应用的导航方法很广泛，包括使用参照坐标系（常用公制）、使用测程法和公制地图。

1.4.2　学习

移动机器人的感知、推理和响应构成了一个闭环，移动机器人因此是研究“智能行为”的理想工具。在自然界中常常可以观察到一种学习的现象——基于经验的行为适应性，这已经越来越多地应用到移动机器人的建模研究中。

关于机器人学习领域的文献是相当多的，这方面的介绍可以参看文献[Franklin, 1996, Dorigo and Colombetti, 1997, Morik, 1999] [Demiris and Birk, 2000] 和 [Wyatt and Demiris, 2000]。

1.5 小结

移动机器人学是一门学科，该学科涉及到如何设计移动机器人的硬件和软件，从而使机器人能够在有噪声、存在不一致和矛盾传感信息，还可能是动态的环境中去执行它们的任务。移动机器人可以接受远程控制、被特殊设计的环境引导（灯塔、条形码、导电环等）或者完全自主运行，例如完全独立而与外部世界无任何连接。

移动机器人广泛地被使用于工业应用场合，包括运输、检验、探索或操作任务。移动机器人在感知和动作之间形成了闭环，因而能够作为研究实现任务（智能）行为的工具。

移动机器人的行为——当机器人运行时所观察到的——发生于机器人、任务和环境之间的交互作用：如果改变机器人的硬件，或者改变控制程序（即任务），或者环境发生了变化，机器人的行为都将随之改变。例如，可以通过改变它的传感器，或者改善它的控制程序，或者在壁面设置多个反射条纹，来改进一个不成功的墙壁沿行机器人。

控制机器人、任务和环境之间的交互作用的基本原理目前只是部分地得到了认识。因此，目前还不可能离线地设计移动机器人控制器，例如，不去检验在目标环境中的真实机器人的情况，通过试错法来进行精细调节。因此，移动机器人研究以及本书的一个目标就是量化地分析机器人、任务和环境之间的交互作用，获得对此交互作用理论上的理解，最终实现离线设计机器人控制器，而且可以对实验及其结果作出量化描述。

第2章 移动机器人学科学方法介绍

概要：本章介绍了全书的主题，区分出目的和目标，并且描述了本书内容的背景。

2.1 引言

移动机器人的行为产生于机器人的控制程序、机器人所处的环境和机器人的物理组成之间的相互关系和交互作用。改变这些组成的任何一部分，机器人的行为都会发生改变。

本书涉及到如何特征化、建模和“辨识”从这三个组成元素之间的交互作用中产生的行为。机器人的行为是否可以预测，它是否可以被建模，它是不是稳定的过程？这个行为是否不同于那个行为，或者它们之间是否没有相当大的差别？哪些程序执行得更好（此处的“更好”是一些可测量指标）？

为了回答这些问题，我们使用的方法来自动力系统理论、统计学和系统辨识。这些方法可以研究机器人—环境交互作用的动力学，然而当这种交互作用也同时被机器人所执行的程序控制时，它们就并不适合用来分析机器人行为的所有方面。例如，对于应用了朝着空间某个特定方向移动的内部模型和计算机视觉的移动机器人，动力系统理论很可能难以充分刻画该移动机器人行为的相关方面。换句话说，本书所说的方法基本上关注的是动力学，而不在机器人行为的辨识方面。

本书旨在延伸自主移动机器人研究的方式，加深研究范围：从一个大量采用反复优化和试错法的学科进步到基于可检验假设的学科，后者可以基于机器人—环境交互的理论而作出有关机器人行为的预测。本书研究改善所观察到的机器人行为的机制：为什么一个机器人会成功适应某些环境而不适应其他环境，能否对机器人将要做什么给出精确的预测呢，能否测量出机器人的行为？

本书所讨论的机制尽管主要关注物理实体的移动机器人及其在真实世界中的操作，但是却能够应用到各种“行为智能体”，可以是软件智能体或动物。在所有例子中下面的问题都是相同的：智能体的行为能否被量化地测量，它能否建模，以及它能否被预测？

2.1.1 讲座计划

本书旨在为本科生和研究生讲授课程“移动机器人学科学方法”，该课程一直在埃塞克斯大学（University of Essex）、纽芬兰纪念大学（Memorial University of Newfoundland）、巴勒莫大学（University of Palermo）和圣地亚哥联合大学（University of Santiago de Compostela）讲授。这些课程的目标是让学生能够对科学研究有一个基本的概念，有助于建立科学哲学[⊖]、实验设计与过程、机器人学与科学分析的相关概念，并将它们特别应用于自主移动机器人学研究的领域。或许按照这些讲座的顺序来介绍本书所涵盖的这些主题是最容易的，实践中讲座效果表明这一顺序是很有效的。

1. 导论（第2章）

1）为什么科学方法与机器人学是相关的？科学方法是怎样应用到自主移动机器人学中的？（2.2节）

2）看作模拟计算机的机器人（2.3节）。

3）机器人—环境交互理论（2.4节）。

4）量化描述的作用（2.4.2节）。

5）机器人工程与机器人科学（2.5节）。

2. 科学方法（2.6节）

1）提出假设（2.6.2节）。

2）实验设计（2.6.3节）。

3）陷阱、缺陷与对策（2.6.3节）。

3. 机器人—环境交互统计描述的介绍

正态分布（3.2节和3.3节）。

4. 比较各种分布的参数检验

1）T—检验（3.3.4节和3.3.5节）。

2）方差分析（3.3.6节）。

5. 非参数检验Ⅰ

1）中值与置信区间（3.4.1节）。

2）曼-惠特尼（Mann-Whitney）U—检验（3.4.2节）。

6. 非参数检验Ⅱ

1）配对观察值的威尔科克森（Wilcoxon）检验（3.4.3节）。

⊖ 从哲学角度考察科学的一门学科。它以科学活动和科学理论为研究对象，主要探讨科学的本质、科学知识的获得和检验、科学的逻辑结构等有关科学认识论和科学方法论方面的基本问题——译者注。

2）克鲁斯卡尔-沃利斯（Kruskal-Wallis）检验（3.4.4节）。
3）随机性检验（3.5节）。

7. 趋势检验

1）线性回归（3.6.1节）。
2）皮尔森（Pearson）的 r（3.6.2节）。
3）斯皮尔曼（Spearman）等级相关（3.7.1节）。

8. 分类数据分析（3.8节）

1）χ^2 分析（3.8.1节）。
2）克莱默（Cramer）的V（3.8.2节）。
3）基于熵的方法（3.8.3节）。

9. 动力系统理论和混沌理论（第4章）

1）相空间（4.2.1节）。
2）移动机器人的自由度（4.2.1节）。
3）机器人学中相空间描述的使用（4.2.2节）。
4）时滞嵌入的相空间重构（4.2.3节）。

10. 基于相空间分析的机器人行为量化分析（4.3节）

11. 吸引子的量化描述子

1）李雅普诺夫（Lyapunov）指数（4.4节）。
2）预测时域（4.4.2节）。
3）相关维数（4.5节）。

12. 机器人—环境交互的建模（第6章）

13. RMAX 建模（6.4.3节）

14. NARMAX 建模（6.5节）

1）环境辨识（6.6节）。
2）任务辨识（6.7节）。
3）传感器辨识（6.8节）。

15. 行为比较（6.9节）

16. 总结和结论（第7章）

2.2 动机：分析机器人学

本书的目的是重点强调这样的问题“当一个移动机器人（或是实际上任何智能体）与它的环境进行交互时发生了什么?”这种交互是否可以作出预测？如果可以建立这样的交互模型，那么这些模型是否能用于离线设计自主移动机器人？就像我们现在不需要开展试错法实验就能够设计建筑物、电路或者化学混

合物。模型是否能够被建立起来以及它们是否能够对交互的性质作出一些假设？机器人—环境交互的过程是随机的还是确定的呢？

为什么上述这些问题是相关的呢？现代移动机器人学是一门相对较新的学科，它使用自主移动机器人，在机器人上配有机载的电源、传感器和计算设备。早在1918年，约翰·海斯·哈蒙德（John Hays Hammond）制造了一个寻光的机器人［Loeb，1918，第6章］。20世纪50年代，格雷·瓦尔特（W. Grey Walter）也制造出通过调节仪器可以学会转向光源的移动机器人［Walter，1950，Walter，1951］。大量的移动机器人学研究直到20世纪80年代才真正开始。就像所有新的学科一样，初始的热点集中在如何让一个机器人工作的工程方面：在特定的任务中可以使用何种传感器，这些传感器的信号需要怎样处理和解释，应该使用哪种控制策略等。所采用的实验方案经常是在相互作用中改进的：首先提出一个关于可行的控制策略的好猜想，然后在目标环境中去检验。如果机器人停住了或不能完成任务，改进控制代码，然后循环上述过程直到能在目标环境中成功完成指定任务。

这种方法得到的结果构成了一种“存在性证明”——某个特定的机器人在一系列特定的环境条件下可以实现一个特定的任务已经得到证明。这些存在性证明是很好的成果，因为它们清楚地表明一种特定的行为或者任务可以实现，但是它们缺乏一个重要的性质：普遍性。机器人在一种环境中能够成功完成一个导航路径，并不意味着它能够在别处也能够完成。而且，实验者不会真正知道为什么机器人会成功。在实验之前不可能以很高程度的确定性指出实验将会成功或者失败。这与桥梁建设不同，建桥梁时土木工程师能够在它还没有被建造前就预测出桥梁的行为，然而机器人学家却不能够在检验前就预测一个机器人的行为。

对于我们来说，或许做出关于在机器人—环境交互中发生了什么的更普遍的、理论化的陈述的时机已经到来。我们有完善的工具，例如计算机模型（参见第6章）和分析方法（参见第4章），我们可以使用它们来建立一个机器人—环境交互的理论。如果这个研究是非实践的，不涉及实体的移动机器人在真实世界中的行为，我有可能称这个学科为“理论机器人学”。而我还是使用了这个词“分析机器人学”。

另外，还可从机器人—环境交互理论中获得许多好处：我们对于机器人—环境交互的理论知识越多，机器人和控制器的设计过程也将更加精确、可靠和低成本。我们对于机器人—环境交互知道得越多，关于实验结果的猜想和预测就会越集中和越精确，进而增强识别出不良实验结果和提高实验设计的能力。最后，对机器人—环境交互的过程理解得越好，就越能更好地分析实验结果，最终实现对实验结果独立重复和验证：机器人学从一种实验性学科前进到拥有

科学方法的学科。

因此，本书的目的是更加清楚地理解机器人—环境交互，并给出关于该交互的抽象的、普遍的表达——机器人—环境交互理论。

2.3　看作执行计算的机器人——环境交互

移动机器人的行为不能孤立地讨论：它是机器人自身（物理方面——“实体”）、环境（所处场景）和机器人所执行的控制程序（任务）三者特性的综合结果（如图 2.1 所示）。机器人、任务和环境的三角形组成了一个复杂的、交互的系统，针对该系统的分析正是关于机器人—环境交互的任何理论的目的。

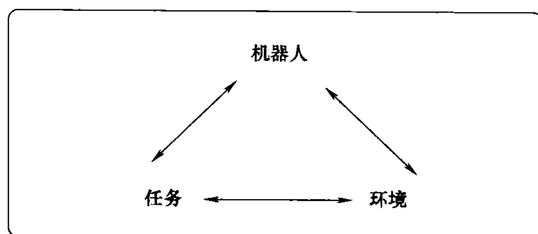


图 2.1　机器人—环境交互的基本三角形

因此，与其孤立地谈到机器人的行为，还不如讨论机器人—环境交互以及由此产生的机器人的行为。

移动机器人，与它所处环境之间的交互，可以看做在执行计算，即一种来自三个输入：机器人形态、环境特征和所执行任务的一种计算行为，如图 2.2 所示。

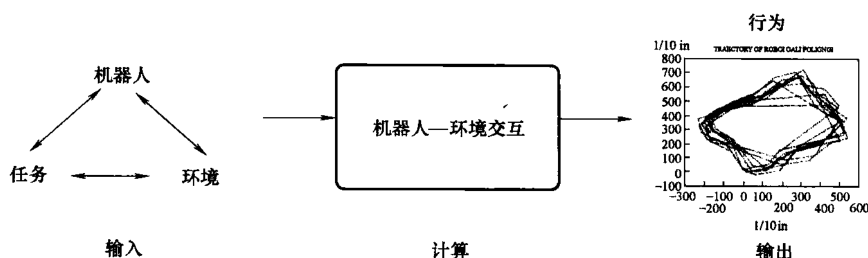


图 2.2　作为计算的机器人—环境交互：行为（输出）是由三个输入：机器人形态、环境特征和所执行的任务的计算得来

类似于一面可以用作模拟计算的柱面透镜，加强垂直边缘的同时弱化水平部分，或者一个通过模拟方法计算傅里叶变换的摄像机镜头，机器人行为——就本书的目的而言，作为一个初始的近似结果，移动机器人的轨迹——可以看做图 2.1 中所示三个组成部分共同作用的结果：机器人从它自身的构成、世界

的构成和考虑进当前运行的程序（任务）出发，“计算”自己的行为。

2.4 机器人—环境交互理论

2.4.1 定义

当谈到“理论”时，我们指的是一种由假设的、概念的和实用的普遍原理构成的统一整体，它形成了开展移动机器人学研究的普遍参考系。

有两个关键因素使得这个机器人—环境交互理论对于研究很有用，因而值得期待：

1）理论应允许对检验提出假设。这是“常态科学”研究中的一个基本组成[Kuhn，1964]。

2）理论将作出预测（例如关于实验的结果），这可以给那些找不到支持或仅有微弱支持的假设予以肯定。

抽象和普遍化提炼中，理论一直保留着由机器人—任务—环境三部分组成的本质。普遍化是基本的：它强调机器人—环境交互的重要方面，而弱化不重要的方面。最终，我们能够评估理论所作出的预测的效果，而得知这种理论是否有效。

对科学学科作出理论上的理解有许多好处。主要的好处在于理论让人们作出一般性假设并给出可检验的预测。同时，它也有实践上的优势，尤其对涉及到人工制品设计的学科。例如，理论支持离线设计，通过使用计算机模型、仿真和基于理论的计算来设计人工制品。

2.4.2 机器人—环境交互的量化描述作用

测量是科学的支柱，它支持：

1）实验设置和实验结果的精确记录。

2）实验参数的原则性的修正。

3）实验结果的独立验证。

4）无需实验开发的人工制品的理论设计。

5）关于所研究系统的行为的预测。

我们已经讨论了机器人行为产生自机器人、任务和环境之间交互作用。假设我们能够量化地测量这个行为。那么，如果图2.1中的这三个组成中的任何两个保持不变，则量化的性能测量能将第三个被修正的成分特征化。这将允许以下的研究：

1）机器人的修正效果。

2）控制程序对机器人行为的影响。

3）环境修正对机器人总体行为的影响。

如图2.3所示：随着一些实验参数（独立变量）的改变，机器人行为（非独立变量）的量化测量发生了改变，因此这可以被用于描述独立变量。例如，对于图2.3中的点γ，量化性能测量有一个全局最大值。

第4章会特别讨论怎样才能用量化的特征来描述机器人—环境交互作用，以及这种使用怎样地量化测量来确定两种影响：①机器人控制器的改变；②环境的改变。

当前移动机器人学研究实践不仅在缺乏支持设计的理论方面与已有学科不同，另一方面：在移动机器人学中实验结果的独立重复性和验证性也很少见。在科学中，例如生物学或者物理学，所得到的结果不仅仅是严格完成了一次而是已经独立验证了多次，然而在机器人学中却不是这样。相反，文献经常陈述的是在特定环境和条件下所获得的实验结果。这些实验因此成为"存在性证明"——证明一个特殊的结果是能够做到的——但是它们没有用一般性的术语说明在何种条件下能够获得某个特殊的结果，或者这个结果是基于何种原理。存在性证明是有用的，它们说明什么事是可以做到的，这也是科学的一个重要方面，但是它们没有提出具有普遍性的原理和理论。

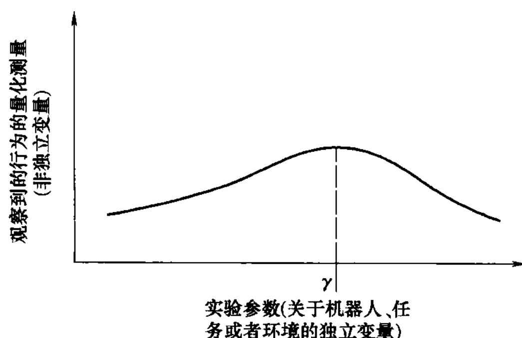


图2.3　量化性能测量和实验参数之间一种可能关系的图示

移动机器人研究现在处于一个应该从存在性证明发展到研究的阶段，这一阶段习惯上包括实验的独立重复和验证。

理论、实验的重复和实验的验证都非常依赖于量化描述：量化描述是科学语言的本质成分。由于这些原因，本书介绍几种量化描述机器人—环境交互作用的方式[㊀]。

㊀ 关于机器人—环境交互的量化测量数据非常丰富，参见文献［Smithers，1995］。

2.5 机器人工程与机器人科学

可以论证，机器人学研究有两个独立的目标：一个方面，为了创造在真实世界中能够执行有用任务的人造装置——例如工业、服务、运输或者医疗的机器人，另一方面为了从理论上理解制造该人造装置的设计问题——例如传感器和驱动器建模、系统辨识（整个系统的建模），或者传感器、驱动器和行为分析。前者可能涉及到“机器人工程”，后者涉及到“机器人科学”。本书主要关注的正是机器人科学。

机器人工程生产“有用的”人工制品的过程中，机器人科学有很大的用武之地。如果没有理论上的理解，任何设计过程将会很大程度上依赖于试错法和反复优化。为了以一种基于原理的方式来设计，就需要一个假设——一个合理预期来指导设计的过程。假设会指导研究：得到的结果反馈到设计过程，并且使其与理论保持一致，进而进入实验和设计的下一阶段。理论为设计过程打下的基础越好，设计过程就会越高效，且越有利于达到目标。即使对所设计物体所知非常有限，每一个设计人工制品的过程仍然都是基于某种假设（即“理论”）。

这同样适用于移动机器人研究。当需要设计一个墙壁沿行机器人时，设计者不会从任意的程序开始进行，而是在开始时，作出一个“合理的猜想”，并明智地假定哪种传感器在实现所期望行为上是有用的，哪种控制程序可以满意地运行等。但是，由于我们当前对机器人学理解的局限，设计者不能在离线情况下设计出全部行为。

相反，移动机器人学研究者们迄今为止非常依赖于试错法的过程。“合理的样机”必须在目标环境中检验，并且基于观察到的现象和基础理论（对于这些理论，“直觉”这个词通常更加合适）再作进一步的改进。这儿有一个实际的例子：为了设计 Roomba 商用保洁机器人吸尘器（依靠非常简单的传感器，且不涉及到任何复杂的导航），在 12 年的期间里一共制造了 30 个样机［EXN，2003］！

然而，对机器人—环境交互从理论上理解将会解决此问题，并且还支持离线设计。这不仅允许我们更进一步分析所观察到的行为，而且可以基于建立的理论原理改进现有的机构。

因此，本书给出的论点是：对移动机器人在其环境中运行的原理理解得越好，理论将带来越有效、越严格和目标导向越明确的开发方法。进而，这些方法将会支持机器人工程，从而开发出更好实现设计任务的机器人。

2.6　科学方法与自主移动机器人

2.6.1　引言

无论移动机器人学是一种科学还是一种工程学科，毫无疑问它都受益于清楚的、连贯的、系统的研究实践，接下来的讨论与“科学”和“工程”都有关。

移动机器人学这一学科关注的是开发能够在真实世界环境中开展某种有用任务的人造装置（机器人）。然而，设计者在采用试错法、系统的研究方法或者两者兼而有之的尝试中，或许不经意间会依赖于某些预先获得的知识。这个知识本质上构成了一个“理论”。细致地分析该理论的组成是什么，以及该理论怎样得到完善是很有用的——这正是本章的目的。

2.6.2　背景：什么是“科学方法”

正如前面所说的，本书的目的是为移动机器人学研究开辟一条新大道，使得原来存在性证明和反复优化的方法转变为能进行离线设计任务实现型机器人的方法。在讨论一些在移动机器人学研究中采用科学方法的实例之前，让我们回顾好几个世纪以来人们对于“科学方法”这个词的广泛理解。这一主题较为恰当的论述可以参看这方面的专业书籍（例如 [Gillies，1996，Harris，1970，Gower，1997]）。

圣·弗朗西丝·培根（Sir Francis Bacon）（1561—1626）首先提出了归纳论 [Bacon，1878]，其基本思想为：首先，收集到关于所研究课题的大量观察数据。这些观察包括“存在事物的例证”、“不存在事物的例证”以及“不同事物的例证”。通过排除归纳的过程可以确定所研究现象的性质。这几乎是很机械化的，通过收集越来越多信息并且排除不可能的假设，真理就被建立起来。文献 [Gillies，1996] 把这种归纳过程比作画精确的圆：只是使用笔和纸是不可能实现的，使用圆规这样的机械装置就很容易实现。科学真理将以一种相似的方式通过归纳的机械过程而被发现。但是，“归纳的问题”在于所收集的事实决不会足够全面到完全可以辨别已经作出的结论，以至于任何假设在效果上仅仅还是假设，这可以说是归纳论的第一个硬伤。

与培根（Bacon）基于许多观察的归纳法完全相反的观点是卡尔·波普尔（Karl Popper）的观点：归纳法是不切实际的，因为没有理论的观察是不存在的 [Popper，1959，Popper，1963，Popper，1972]。换句话说，为了观察，需要先有一个理论，因而仅靠大量观察就获得理论是不可能的。科学家不可能进行简单的“观察”，必须知道应该观察什么。进而，还需要定义所选择的任务、问题

和困难——也可以称为假说。与归纳论不同，他提出了一种演绎与证伪的理论（证伪主义）：科学研究的目的是证伪一个假说，所有实验都应朝着这个目标进行。如果一个假说经受住了各种证伪的尝试，它暂时被认为是正确的，但并不认为它已是经过证明的毫无疑问的真理。能够永远成立的唯一真理就是一个理论是假的，而决不是它是真的。

那么科学共同体如何接受或拒绝某个理论呢？托马斯·库恩（Thomas Kuhn）[Kuhn，1964] 区分出“常态科学”和“科学革命”的一种情形。他所描述的常态科学的研究是牢固地基于过去的科学成果或者“范式”的。这里的范式指那些能给出科学发现途径、形成问题、确定方法和定义相关研究领域的理论——范式可以指导研究。“常态”科学研究的目的在于在现有范式内使知识得到延伸，让理论与事实相符合，清楚地说明理论，使它与所观察到的事实更一致。这往往抑制了与现存范式不一致的新理论的产生。在已被接受的现有范式范围内，常态的科学工作是探求扩展该范式已有的知识，“捆扎松开的地方”，并堵上漏洞——库恩把这个叫做“修修补补”。

然而，在常态科学发展的过程中，我们观察到事实与（非常态的）理论之间的矛盾不断增加。总会有一些不能够被现有理论完全解释的观察结果，总会有一些与现有理论不一致的观察结果。这些矛盾的情况最初往往被忽略，但随着它们重要性的增加，直到有一天科学共同体不再信任现有的范式，一场危机产生了；最初是现有范式的模糊化，接着产生一个改变范式的提议，进而导致“科学革命”，从“常态”过渡到异常研究。最终，新的范式被大多数的科学家承认，并且被看做“常态”范式，然后此过程又重复下去。

1. 科学研究方法论

正如引言中所说，本书不是介绍科学哲学和它在移动机器人上的应用的。当我们提到“科学方法”时，重点不是关于研究方法论的哲学基础。

相反，本书的重点是关于开展、评价和记录研究及其结果的过程；即科学实践的内容。对于研究来说什么才是一个好的起点？怎么设计实验，怎么记录和评价结果？我们在共同体范围内采用什么作为科学研究步骤？在开展实际研究之前我们应该能够回答以上各种各样的问题。

2. 提出科学假设

所有研究的起点都是提出一个假设、一个论题。这个假设是关于定义了研究目的和目标的行为的正式陈述的期望；因此它定义、解释和指导着研究。如果在最开始没有一个清晰的假设，开展好的研究实际上是不可能的，同样地，以一贯的和令人信服的方式给出结果，实质上也是不可能的。假设，即问题，是建立起科学论题的基础。显然，一个模糊不清的问题将导致模糊不清的回答，这也正是提出假设是科学工作的最基础阶段的原因。

为了清楚地提出假设，考虑以下几点是很有用的（可以参看［Paul and Elder，2004］）：

（1）所要解决的问题是什么？

1）准确地提出问题

2）该问题是否能够被拆分为若干子问题？

3）此问题是否有一个正确的答案？该问题需要来自多于一个观点的推导吗？它是不是因为看法不同带来的问题？

（2）你做了什么假设？

1）清楚地区分所有的假设

2）这些假设是有理由的吗？

3）这些假设影响了研究的客观性吗？

4）区分形成研究的关键概念和想法。它们是合理的吗？

（3）设定假设

1）假设可被检验并证伪吗？

2）你期望什么结果？

3）实验的不同可能的结果的含义是什么（比如，是确实值得问的问题吗）？

4）实验的设计

（4）哪一个实验设计适合研究这个问题/假设？

1）怎样采集实验数据？

2）怎样评价实验数据？

3）需要多少数据？

假设可以是因果型假设，关于一个行为产生原因的假设，或者叙述型假设，根据行为的特点或者它所处的环境来描述该行为。因果推理和因果关系模型在科学中是很常见的，它们能指导实验设计、假设和理论的形成。因果关系模型对科学思考指导的影响之大，以至于有时科学家们甚至不顾他们所得到的统计信息而青睐因果模型［Dunbar，2003］（其中谈到的“证实偏见”——“20 世纪 80 年代末冷核聚变事件”就是个突出的例子）。换句话说：指导研究的假设的主导作用是如此明显，以至于科学家力图得到证实他最初假设的结果，而不是旨在反驳这个假设（根据波普尔的观点，这正是他应该做的！）［Klayman and Ha，1987］——试图证实假设而不是证伪它的倾向是很难被克服的。做能得到由当前假设预测出的结果的实验的倾向是很强的！

波普尔主张（由于宇宙无限大）科学假设永远不可能被证实（即被证明是正确的），它的真实性也无法证实，而只能被证伪（即被证明是错误的）。而且，他主张对于任何科学假设的基本要求是该理论对检验、修正是无条件的。换句话说：它必须是可检验的，并且它必须是可证伪的。如果这些条件之一不满足，

那么这个假设将无法支持科学研究。

波普尔注意到通过采用“补救策略”（例如通过修改假设的可检验性）可以规避证伪，因此他引入最高原则“科学程序的其他原则必须以这样的方式进行设计：“它们不会让科学中任何陈述免于被证伪”［Popper，1959，p. 54］。

“科学的目的是找到满意的解释，而不在乎这种解释会对我们造成怎样的打击”［Popper，1972，p. 191］——研究中的假设最终能确定某个解释令人满意或不满意的程度。

区别“好的”假设与“坏的”假设还有更进一步的标准。波普尔和库恩（Kuhn）将解释深度作为关键方面——哪一个范式可以解释更多的现象？而增加的真实性同样被波普尔当做建立假设的目标。在库恩、拉卡托斯（Lakatos）和劳丹（Laudan）所写的一篇综述文章中［Nola and Sankey，2000］，总结了各种观念得出“科学家更喜欢这样的理论”：

1）面对竞争者能够解决一些经验性的问题。

2）能够将明显的反例变成可解决的问题。

3）能够解决那些原本没有打算解决的问题。

4）能够解决那些前人没有解决的问题。

5）能够解决前人已经解决的所有问题，还多解决了一些新问题。

6）能够解决大量重要的经验问题，同时只产生最少的例外和概念困难。

假设必须精确、合理（即可能正确并且与已知相一致）并且精炼的（即尽可能简单——而不是较为简单）。奥卡姆的剃刀（Occam's razor）[⊖]——“如无必要，勿增实体（entities are not to be multiplied beyond necessity）”——正是表达了这一原则。总结来说，一个“好的”科学范式的特点——它必须是可以被检验和证伪的——是有解释能力、清楚并且一致的。

科学假设如何获得呢？最常见的来源是：

1）观点、观察和经验。

2）现有的研究。

3）理论。

4）模型。

卡尔·波普尔主张科学假设是科学家大量创新思考的产物（他将此称为“创造性的直觉”）。

⊖ 又称“奥康的剃刀”，是由14世纪逻辑学家、圣方济各会修士奥卡姆的威廉（William of Occam）提出。他在《箴言书注》2卷15题说“切勿浪费较多东西去做用较少的东西同样可以做好的事情”——译者注。

2.6.3 实验设计和过程

1. 实验设计

实验设计——所采取的实验过程、观测的机制和解释结果的方式——是所有科学研究的中心，设计实验时需要小心仔细。所选择的设计是否适合研究所感兴趣的问题呢？是否有实现目标的更好方式呢？设计的实验是否在实践中切实可行，或者说它是否会产生不可克服的实际问题？

科学实验的最通常类型之一是旨在确定两个变量之间的关系：一个由实验者所控制（独立变量，IV），一个则是依赖于实验者（非独立变量，DV）。实验的最通常目的是为了确定非独立变量（DV）随着独立变量（IV）如何改变。

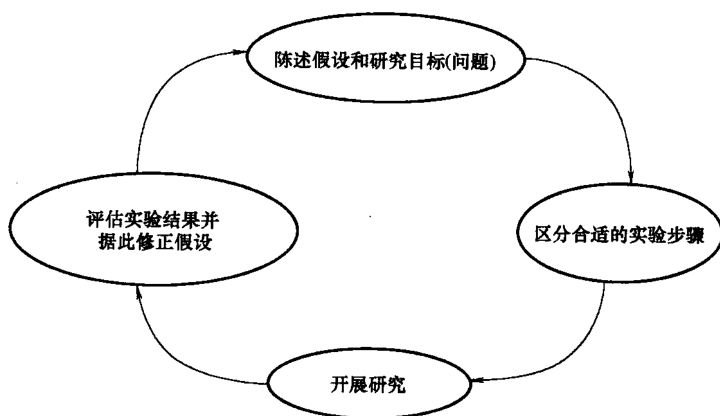


图2.4 实验步骤

设计实验时需要考虑许多方面，如图2.4所示：

- 问题是什么？假设是什么？每一项科学研究都是由基础的要回答的问题所驱动。如果这个问题没有被清楚地陈述，或者甚至完全没有表达，它所产生的研究将是随意的、病态的，不能集中在清晰目的上。好的研究需要一个清楚表达出来的问题！
- 实验的灵敏度。实验设计是否足够灵敏到可以检验出非独立变量（DV）和独立变量（IV）之间任何因果关系？或者由于它太灵敏反而将噪声也放大了呢？
- 是否存在由于引入误差产生的混淆效应掩盖了真实效果的问题（参看下面关于这点的讨论）？
- 对于实验结果我们能够作出哪些预测？基于问题的一些预先理解，是否存在某些期望能被用于评价实验的最终结果？

预测是重要的，利用它们作出合理性检验，帮助我们找出那些极不可能的结果，检验实验设计、过程和评价中可能的错误。

- 观察。实验将怎样被观察，结果如何记录？此处尽可能连贯起来是很重要的。相似的实验应该导致相似的结果，如果不是这样，那么我们应该再检查一下实验设计，看看是否有可能的误差来源或者步骤上的错误。
- 分析和解释。结果如何分析？用语言描述实验结果只是一种可能，我们还可使用功能更强大的工具进行分析。第 3 章包括了可以用来检验“显著性”效果的全部统计学方法。

用于分析和解释的一个非常重要的方法是找到一个基线。它是人们比较实验结果的一个基本参照。

例如，在与预测相关的工作中（例如，预测机器人轨迹、室外的温度或者股市的发展），人们经常用预测方法与预测均值的基线来比照。未来信号值是过去值的均值的这种预测是一种非常合理的预测，这种预测往往预测误差最小。如果一个预测方法被认为是“好的”，它应该胜过刚才说的这种简单的预测方法——采用这些方法可建立什么将在第 3 章中介绍。

- 为了减小实验的工作强度，先开展一些试验性的研究通常很有用。在较大范围内，针对主要问题进行试验性研究可以减少某些可能性，使得其他可能性更大。这种试验往往比最终的实验在实验程序上更简化和更粗略。

2. 陷阱与对策

（1）陷阱　有许多需要避免的陷阱 [Barnard et al，1993]：

1）混淆效应。如果我们感兴趣的现象与我们不感兴趣的其他某个效应之间有很大的关系，那么必须特别小心地设计实验，仅研究感兴趣的因素。例如，我们或许会对移动机器人的避障运动是否会变得更有效、更稳定感兴趣。我们或许会发现我们的学习算法是能够使机器人更稳定运动的正相关因素。但是，除非我们非常小心地设计实验，否则我们便不能确信这种变得更加稳定的现象不是由于电池电压而导致机器人的电动机响应变缓所致。

2）下限效应和上限效应。有可能实验设计得要么过分苛刻，要么过分简单而使得相关现象没能得到加强。例如，我们有可能对研究一个服务机器人是否比另一个运行得更好感兴趣。如果我们在过分简单的环境中比较两个机器人，它们可能不会表现出任何不同（下限效应）。另一方面，如果我们选择一个非常复杂的环境，则没有一个机器人可以在其中满意地运行（上限效应）。显然，为了强调两个机器人的不同，需要采用有恰当复杂性的环境类型。

3）假重复（非独立）。一个实验的成本（就时间或资源而言）越高，产生非独立（所谓的假重复）数据的风险就越大。假重复意味着我们测量的误差对每次测量来说都不是唯一的，即是不独立的。

例如，我们可能想要测量机器人用它的摄像机检验物体颜色的效果。比如，我们可以用三个不同的物体做实验，让机器人对每个物体检验十次。但这不需要做三十个独立测量！这种情况下，我们实际只需做三个测量，在我们进行任何结果分析之前需要将每个物体的十个观察值缩减为一个。

4）固有误差是系统误差（偏差），能够掩盖真实效果因而需要避免。

5）“善意的陷阱”（彼得·梅达沃，Peter Medawar）——在设计实验中，我们需要非常小心地保持目标。我们的实验很容易在心里存在期望的结果，很容易有所选择地进行研究而达到那个结果。

（2）对策 有若干对策可以避免上述提到的缺陷。

首先，很好的实践是在实验设计中引入控制。这样的控制可以采用设置给定并且结果已知的实验形式。举个例子，设计一个探险机器人，用于在某些遥远地区（如比南极）探测某些物体（如石块）。控制的通常步骤是在机器人的探测能力可以被观察和测量的实验室环境中，检验机器人和它探测物体的能力。

其次，克服科学研究缺陷经常使用和非常有效的方法是分组工作，以便找到对某组的实验设置、步骤、结果和解释的独立验证与证实。通常假设、设置和解释得益于仔细的观察。

通过抵消平衡法和随机化法可以避免固有误差。抵消平衡法代表着一种实验过程：研究中每个变量都使用相同的次数。例如，如果在同样的环境中检验两个不同的机器人控制器A和B，平衡法的实验设计是指A和B分别在第一次和第二次使用相同的次数。这一作法会抵消掉损耗（比如电池电量减少）带来的固有误差。

处理固有误差的另一方法是随机化法。随机化法相当于用偶然性来进行误差平衡：变量的使用是随机决定的。

平衡法只能用于所平衡的变量之间没有相互作用时。举例说明，如果上例中的程序B改变了环境，比如重新布置环境中的物体，程序A和B以何种顺序被检验就有关系了。这种情况下平衡法将无法起作用。

“善意的陷阱”的处理相对容易：“盲目的”实验安排就可以达到。盲目的实验意味着实验者并不知道独立变量的状态，因而以表面值来记录和解释得到的实验数据，而不是在解释中不经意间加入某些倾向。

药品试验中众所周知的科学问题是某种药物是否有效（病人和医生都不知道正在服用的是药物还是安慰剂）。这种盲目的实验实际上在机器人学中也占有一席之地。实验者总存在那种在试图解释结果或者拿自己的控制程序与基本控制程序对比的偏好。如果实验者不清楚正在运行哪个程序，那么他就不可能在记录和解释数据时抱有偏见。

2.7 本书所用的工具

2.7.1 Scilab

在本书的某些章节中，涉及了很多文中讨论的方法和算法的例子。我们使用数学程序包 Scilab [Scilab Consortium, 2004] 来说明例子，还包括一些程序集。本书的许多图是使用 Scilab 生成的。

Scilab 是一种功能强大的数学编程语言，另外一个优点就是它对于个人使用是免费的。本书所给例子用其它数学编程语言运行时也几乎不需要修改，例如 Matlab。

2.8 小结：实验的移动机器人学和科学的移动机器人学的对比

综上所述，作为实验的移动机器人学和作为科学的移动机器人学之间的对比可以描述为以下几点：

1）实验设计和过程是由可检验的、可证伪的假设所指导的，而不是基于研究者的个人经验（直觉）。

2）实验设计和过程是由“问题驱动”，而不是由“应用驱动”。

3）结果是定量地而不是定性地进行检验和分析。

4）实验结果可以独立重复和验证（例如由其他研究小组进行），而不是仅仅给出一个孤立的证据。

本书后面的章节将介绍如何达到这些目标。怎样评价移动机器人的性能，怎样比较两个控制程序？如何定量地描述机器人—环境交互？如何提出可检验的假设？怎样建立机器人—环境交互的模型并进行精确的仿真？这些正是我们现在要开始研究的问题。

第3章　描述实验数据的统计工具

概要：实验数据的统计描述是量化描述机器人行为的最简单方法之一。本章所介绍的统计方法可以用于分析智能体，如一个机器人所产生的实验数据，并给出了大量机器人学方面的应用实例。

3.1　引言

本章介绍用于分析、描述或量化机器人行为的大量统计技术。它们很多来自有完善的统计分析和行为比较的生命科学。

在任何实验科学中，如生物学、心理学、医药学或机器人学等，我们通过所设计的实验来检验我们的假设。观察实验，记录描述实验相关方面的数据，然后进行分析。一旦记录了实验数据，我们就需要回答以下这些问题：

1）输入和输出变量之间存在统计上显著的相关性吗？

2）所得到的实验结果与某个“基线”（可以是实现同样任务的其他方法，也可以是已被较好认识的某种方法）之间是否存在统计上的显著差异？

3）作为选择，实验结果能被随机事件解释吗？是一个偶然事件吗？

此处“统计上显著的”是一个被精确定义的术语，是指实验结果不同于“无效假设”[⊖]，不是随机波动的结果。本章稍后将讨论“显著性”这个词。

统计学检验的步骤：1）确定需要分析的数据类型，2）确定提出的是何种问题（是否是无效假设），3）选择一种合适的检验，4）进行分析。

本章先介绍将要分析的数据类型（正态分布或非正态分布），并给出一些确定两个样本是否来自同一个总体分布（例如，它们彼此是否存在显著差异）的方法，然后确定是否存在能描述两个变量关系显著性趋势的检验方法，最后给出适合分析分类数据（数据通常不是数字的，而是基于类别的）的方法。表3.1给出了本章所讨论的统计方法。

表3.1　本章所讨论的统计方法

	样本是否来自同一分布	趋势检验
数据服从正态分布	均值和标准差（3.2节） t—检验（3.3.4节） 参数化方差分析（3.3.6节）	线性回归（3.6.1节） 相关分析（3.6节） 斯皮尔曼（Spearman）等级相关（3.7.1节）

⊖ 指观测的实验结果只是由随机事件而不是系统原因引起的假设。

（续）

	样本是否来自同一分布	趋势检验
数据服从正态分布	U—检验（3.4.2节） 威尔科克森检验（3.4.3节） 非参数方差分析（3.4.4节）	线性回归（3.6.1节） 相关分析（3.6节） 斯皮尔曼（Spearman）等级相关（3.7.1节）
数据服从非正态分布	中值（3.4.1节） U—检验（3.4.2节） 威尔科克森检验（3.4.3节） 非参数方差分析（3.4.4节）	斯皮尔曼（Spearman）等级相关（3.7.1节）
分类数据		χ^2（3.8.1节） 克莱默（Cramer）的V（3.8.2节） 熵（3.8.3节）

3.2 正态分布

高斯（Gaussian）分布是测量值（如测量物体得到的）最常用的分布。它是经常发生的分布，通常称为“正态分布”。

1. 均值、标准差、标准误差

正态分布可以完全地由两个参数：均值μ和标准差σ描述。对于正态分布，所有测量x_i值中的68.3%位于区间$\mu \pm \sigma$，所有x_i中的95.4%位于区间$\mu \pm 2\sigma$，有x_i中的99.7%位于区间$\mu \pm 3\sigma$。在正态分布中，数值的分布以“期望值”即“均值μ”为中心，钟型曲线的宽度由所谓的“标准差σ”决定，下面会谈到更多这方面的内容。

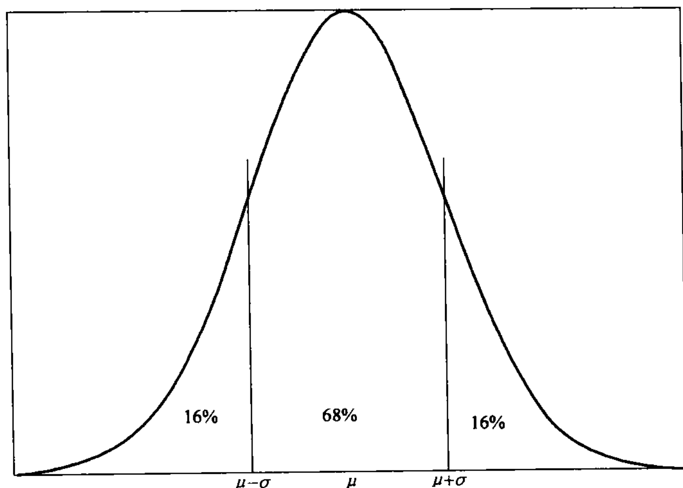


图3.1 正态分布。数值分布在以均值μ为中心的附近，并且离均值越远的数值出现的频率越低；所有测量值中的68.3%位于区间$\mu \pm \sigma$

正态分布的概率密度 $p(x)$，如图3.1所示，由式（3.1）定义。

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{3.1}$$

测量的期望值——均值 μ 由式（3.2）定义。式（3.3）[⊖]给出了标准差 σ。

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{3.2}$$

这里 x_i 是一系列测量中的一个单次测量值，n 是测量的总数。

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\mu)^2} \tag{3.3}$$

均值和标准差充分地描述了正态分布，包含了测量值 x_i 准确度的信息，其中，σ 表明有多大比例的测量值落入特定的区间 $\mu \pm k\sigma$。例如，当 $k=1$ 时，这个比例是68.3%。

对于每个单次测量值 x_i，均值本身也有误差，我们可以确定均值的均值误差，$\bar{\sigma}$为

$$\bar{\sigma} = \frac{\sigma}{\sqrt{n}} \tag{3.4}$$

这个所谓的标准误差$\bar{\sigma}$是对均值不确定性的测度，$\mu \pm \bar{\sigma}$表示真均值以68.3%的确定性落在该区间；$\mu \pm 2\bar{\sigma}$表示真均值以95.4%的确定性落在该区间；$\mu \pm 3\bar{\sigma}$表示99.7%的置信区间。当文献中陈述均值或以条形图表示时，我们通常是指$\mu \pm \bar{\sigma}$。

减少测量误差

随着测量次数的增加，标准误差$\bar{\sigma}$减小。这意味着由真均值得到的均值 μ 的偏差也减小。然而，当标准误差正比于单次测量的均值误差（式3.3），反比于$\sqrt{n}$，随意地增加测量次数，来减少不确定性就不是很有用。如果想要减少测量误差，更好的办法是提高测量精度。

2. 标准正态分布

正态分布对衡量很多统计实验的结果起着重要作用。通常当 $\mu=0$ 和 $\sigma=1$ 的标准正态分布用在以下这类对比中；表3.2给出了这种分布，$\mu+z$ 区间正态分布曲线下面的面积（见图3.2）。

任何均值不为零、标准差不为1的正态分布都可通过式（3.5）变形为标准正态分布：

$$z(x) = \frac{x-\mu}{\sigma} \tag{3.5}$$

⊖ 式3.2和3.3是实际中使用的近似公式。严格地说，$\mu = \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}x_i$ 和 $\sigma = \lim_{n\to\infty}\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\mu)^2}$。

表 3.2 当$\mu=0$和$\sigma=1$时，区间μ到$\mu+z$的标准正态分布表

z	0.00	0.02	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.000	0.004	0.008	0.012	0.016	0.020	0.024	0.028	0.032	0.036
0.1	0.040	0.044	0.048	0.052	0.056	0.060	0.064	0.067	0.071	0.075
0.2	0.079	0.083	0.087	0.091	0.095	0.099	0.103	0.106	0.110	0.114
0.3	0.118	0.122	0.126	0.129	0.133	0.137	0.141	0.144	0.148	0.152
0.4	0.155	0.159	0.163	0.166	0.170	0.174	0.177	0.181	0.184	0.188
0.5	0.191	0.195	0.198	0.202	0.205	0.209	0.212	0.216	0.219	0.222
0.6	0.226	0.229	0.232	0.236	0.239	0.242	0.245	0.249	0.252	0.255
0.7	0.258	0.261	0.264	0.267	0.270	0.273	0.276	0.279	0.282	0.285
0.8	0.288	0.291	0.294	0.297	0.300	0.302	0.305	0.308	0.311	0.313
0.9	0.316	0.319	0.321	0.324	0.326	0.329	0.331	0.334	0.336	0.339
1.0	0.341	0.344	0.346	0.348	0.351	0.353	0.355	0.358	0.360	0.362
1.1	0.364	0.367	0.369	0.371	0.373	0.375	0.377	0.379	0.381	0.383
1.2	0.385	0.387	0.389	0.391	0.393	0.394	0.396	0.398	0.400	0.401
1.3	0.403	0.405	0.407	0.408	0.410	0.411	0.413	0.415	0.416	0.418
1.4	0.419	0.421	0.422	0.424	0.425	0.426	0.428	0.429	0.431	0.432
1.5	0.433	0.434	0.436	0.437	0.438	0.439	0.441	0.442	0.443	0.444
1.6	0.445	0.446	0.447	0.448	0.449	0.451	0.452	0.453	0.454	0.454
1.7	0.455	0.456	0.457	0.458	0.459	0.460	0.461	0.462	0.462	0.463
1.8	0.464	0.465	0.466	0.466	0.467	0.468	0.469	0.469	0.470	0.471
1.9	0.471	0.472	0.473	0.473	0.474	0.474	0.475	0.476	0.476	0.477
2.0	0.477	0.478	0.478	0.479	0.479	0.480	0.480	0.481	0.481	0.482
2.1	0.482	0.483	0.483	0.483	0.484	0.484	0.485	0.485	0.485	0.486
2.2	0.486	0.486	0.487	0.487	0.487	0.488	0.488	0.488	0.489	0.489
2.3	0.489	0.490	0.490	0.490	0.490	0.491	0.491	0.491	0.491	0.492
2.4	0.492	0.492	0.492	0.492	0.493	0.493	0.493	0.493	0.493	0.494
2.5	0.494	0.494	0.494	0.494	0.494	0.495	0.495	0.495	0.495	0.495
2.6	0.495	0.495	0.496	0.496	0.496	0.496	0.496	0.496	0.496	0.496
2.7	0.497	0.497	0.497	0.497	0.497	0.497	0.497	0.497	0.497	0.497
2.8	0.497	0.498	0.498	0.498	0.498	0.498	0.498	0.498	0.498	0.498
2.9	0.498	0.498	0.498	0.498	0.498	0.498	0.498	0.499	0.499	0.499
3.0	0.499	0.499	0.499	0.499	0.499	0.499	0.499	0.499	0.499	0.499

x是原正态分布的一个值，它可以转换成标准正态分布值。

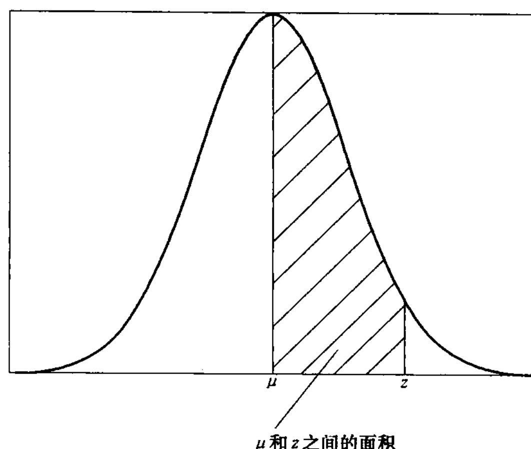


图 3.2　表格 3.2 给出了位于正态分布的均值 μ 和 z 之间的面积占曲线下整个面积的比例

表 3.2 可以用来确定分布曲线下的面积。例如，根据此表，坐标 μ 到 1 与曲线围成区域的面积是 0.341。因为标准正态分布曲线下区域的整个面积是 1，这就意味着 34% 的数据点落在 $\mu+1\sigma$，或者说 68% 的数据点落在 $\mu\pm\sigma$。

3.3　样本比较的参数方法

3.3.1　一般考虑（显著性水平）

在进行统计检验时，主要是要确定检验统计量 S，并估计 S 是否落在接受区间 $\mu\pm z\sigma$，其中 z 由用户指明的显著性水平决定。这个显著性水平表示了由上下分布曲线围成的面积（图 3.3 中的阴影部分）。由此确定了接受区间的宽度。

如果样本统计量落在接受区间外面，那么这个无效假设[⊖]就被拒绝，否则就被接受（这仅仅意味着没有有效的理由否定它，但却不一定它是真的）。

样本统计确定合适的显著性水平。接受区间显然对一个无效假设是被接受还是被拒绝有影响：接受区间越宽（也就是显著性水平越低），无效假设被接受的可能性就越大。

在选择合适的显著性水平时，我们要考虑是否在特殊的情况下否定一些可能是正确的无效假设是恰当的（Ⅰ型错误——弃真错误，显著性水平太高），或者接受一些错误的无效假设（Ⅱ型错误——取伪错误，显著性水平过低）。这些

⊖ 例如，两个分布之间没有显著差异。

错误是相互依存的；我们在选择其中一个变小时，那么另外一个就会变大。合适的显著性水平通常的代价是两种错误将同时出现。在科学研究中，显著性水平一般设为5%。

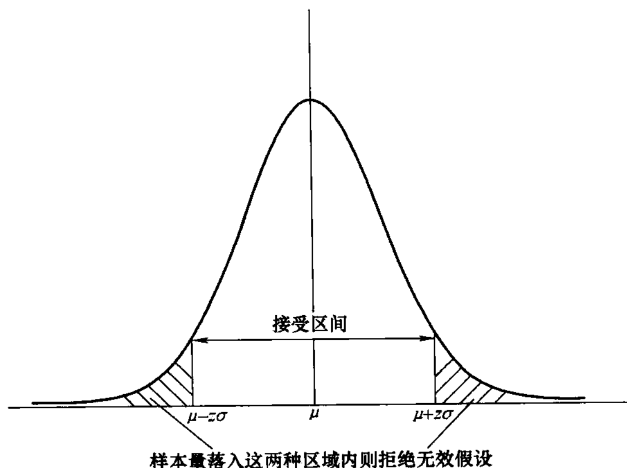


图3.3 显著性水平的例证。如果样本统计量落在接受区间外，无效假设被拒绝。

举例

我们用$\mu_K=50$，$\sigma_K=6$时的一个正态分布的样本统计量来估计一个实验的结果。请问当显著性水平分别是5%和10%时的接受区间是多大?

从表格3.1我们知道$z=1.96$时的面积是47.5%（$2\times47.5\%=95\%$）。显著性水平为5%时的接受区间是$\mu_K\pm z\sigma_K=50\pm1.96\times6$（也就是38.2～61.8）。

显著性水平为10%时，从表格3.1我们可以得到$z=1.65$，计算接受区间：$50\pm1.65\times6=40.1\sim59.9$

因此，假设一个样本统计量为61，我们就可以接受显著性水平为5%的无效假设，而拒绝显著性水平为10%的无效假设。

在所有参数方法中最基本的假设是所分析的数据服从正态分布——参数方法只能用于已知分布状态的数据。因为这个限制，参数方法与非参数方法相比显得不够“强大”，遇到不服从假定分布（一般是正态分布）的数据的话，会得出错误的结果。但另一方面，它可以比非参数检验方法更好地拒绝一些无效假设，这个特点经常被认为更“强大”。

3.3.2 确定一个分布是否是正态分布

如上所述的情形，所有参数检验都假设基于的数据分布是（或接近）正态分布。然而，它经常不被认为是一个先验，如果我们想要用参数检验方法，就

必须研究这些数据能不能用一个正态分布很好地表示。

当样本容量很小（$n < 10$）时，经常很难确定数据点是否服从某一个正态分布。因为如果这些数据服从其他某个分布，那么用为正态分布的数据而设计的检验就会得出错误的结果，这时建议使用非参数检验方法，因为非参数检验方法没有假定总体分布。

当样本容量 $n > 10$ 时，那么判断总体分布是不是正态分布最简单的方法是用人眼比较数据的直方图和正态分布曲线是否吻合。图 3.4 列出的是图 4.12 中机器人避障行为的直方图。只用裸眼就能很明显地看到直方图分布状态比正态分布多一个峰值（两个峰值）。

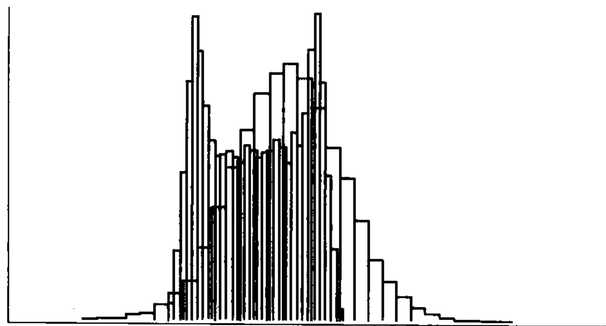


图 3.4　机器人避障直方图与正态分布直方图的比较

一种更精确的绘图方法是将数据画在正态分布概率纸上，它上面的坐标系可以让服从正态分布的数据落在一条线上。然后我们就可以用眼睛直接分辨某个数据与曲线吻合得好不好。

在 Scilab 中，将数据点标在正态分布概率纸上有两种等效的命令（说明：第一条命令有三行）：

```
plot2d([0 sum(abs(a) <0.25)sum(abs(a) <0.5)sum(abs(a) <1)
sum(abs(a) <2)sum(abs(a) <3)  length(a)],
[0  20  38.3  68.2  95.4  99.7  100])
xpoly([0  length(a)],[0  100])
```

其中，a 是所分析的样本。图 3.5 表示的是利用这种方法画出的正态分布随机数列图和均匀分布的随机数列图。

3.3.3　样本是独立的还是相关的?

一些检验的对象是“独立的”或“相关的”样本。“相关的”样本就是配对的数据，例如一些机器人在修正之前或之后所执行的动作。事实上，如果你不能在不丢失信息的前提下将两组数据打乱，样本就是相关的。如果你将配对

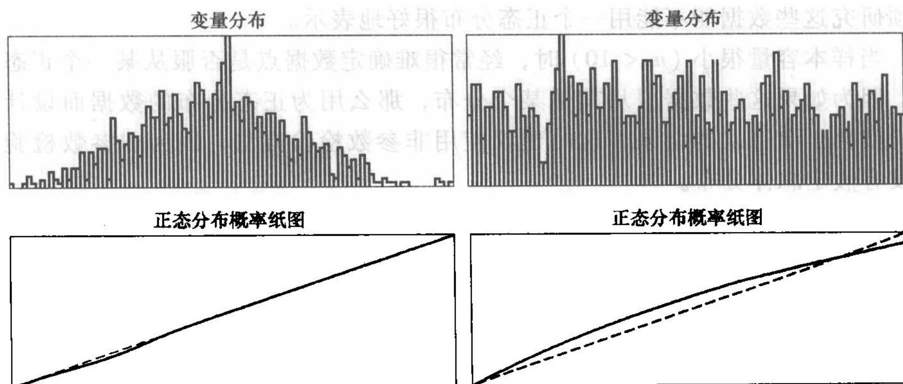


图 3.5　正态分布随机数列的直方图和画在正态分布概率纸上的图（左图），均匀分布随机数列的直方图和概率纸图（右图）。对于正态分布，在正态分布概率纸上的结构图是一条直线，而对于其他任何分布情况就不是这样了

的数据（如一些前后相关的数据）打乱，你将会丢失一些至关重要的信息，那么这种数据是相关的。从另一方面讲，将一种类型的机器人数据打乱，再将另一类型的机器人数据打乱，不会丢失信息，这种数据就是独立的。

另一种描述配对数据的方法是分析者关心的，不是两者的统计学特性，而是两个样本间的不同点。

3.3.4　比较两个样本：用于分析独立样本的 t-检验

对某种特定的算法、机构控制等性能的测量通常是很有用的。例如，如果两个不同的控制程序对某个特定结果产生两种不同均值，这就很有必要去判断这两种均值间有多大的显著差异，以确定两个程序中哪一个产生的结果更好。

t-检验是用来比较正态分布值中的两个均值 μ_1 和 μ_2 的，它们的标准差（几乎）相等。无效假设的 H_0 为 $\mu_1 = \mu_2$。t-检验所用的统计学测量值是式（3.6）中的 T；如果 T 值在接受区间外，那么无效假设的 H_0 被拒绝：

$$T = \frac{\mu_1 - \mu_2}{\sqrt{(n_1 - 1)\sigma_1^2 + (n_2 - 1)\sigma_2^2}} \sqrt{\frac{n_1 n_2 (n_1 + n_2 - 2)}{n_1 + n_2}} \tag{3.6}$$

式中，n_1，n_2 分别是试验 1 和 2 的数据点个数；μ_1，σ_1 是试验 1 的均值和标准差；μ_2，σ_2 是试验 2 的均值和标准差。

检验这样进行：由表 3.3 确定，令 $k = n_1 + n_2 - 2$ 是自由度数（这个表格给出了所谓的“双尾检验”的临界值，也就是检验任一种方法是否显著大于或小于另一个。换句话说，就是检验统计学测量值 T 是否落在分布曲线的两端）。如果不等式 $|T| > t_{crit}$ 成立，则无效假设 H_0 被拒绝，表示两种均值有显著差异。t-

检验结果是错的概率表示了当概率为0时，显著性取决于所选择的显著性水平（参照3.3.1节）和临界值 t_{crit}，反之亦然。通常，我们取显著性水平值为5%（$p=0.05$），但取决于是否Ⅰ型错误或Ⅱ型错误越少，哪种显著性水平更合适。

表3.3　显著性水平分别为2%，5%和10%，双尾检验的临界值 t_{crit}，k 为自由度

k	1	2	3	4	5	6	7	8
$t_{0.02}$	31.821	6.965	4.541	3.747	3.365	3.143	2.998	2.896
$t_{0.05}$	12.706	4.303	3.182	2.776	2.571	2.447	2.365	2.306
$t_{0.10}$	6.314	2.920	2.353	2.132	2.015	1.943	1.895	1.860
k	9	10	14	16	18	20	30	∞
$t_{0.02}$	2.821	2.764	2.624	2.583	2.552	2.528	2.457	2.326
$t_{0.05}$	2.262	2.228	2.145	2.12	2.101	2.086	2.042	1.960
$t_{0.10}$	1.833	1.812	1.761	1.746	1.734	1.725	1.697	1.645

用现有的商业软件包可以很容易地算出 T 分布，而不是用像表3.3这样的表格。例如在Scillab软件中，编辑下面一段语句可以计算出 t_{crit}

[T] =cdft（“T”，k，P，Q）

这里的k是自由度，Q相当于希望得到的显著性水平（因为我们在用双尾检验），P=1-Q

确定一下，例如 t_{crit} 中有k=19，显著性水平是5%，我们可以得到：

[T] =cdft（“T”，19，0，0.975，0.025）

T=2.0930241

t-检验的实例：从死角中退出（Dead End Escape）

一个机器人的控制程序是为了让机器人能从死角中退出。在程序的第一版中，机器人用了下面的几秒钟时间来从死角中逃离。

Exp1 =（10.2，9.5，9.7，12.1，8.7，10.3，9.7，11.1，11.7，9.1）

程序被改进后，实验得到另一批结果：Exp2 =（9.6，10.1，8.2，7.5，9.3，8.4）。

把这些结果记录到表3.6中。它们能说明第二个程序明显更好吗？

答案：假设实验的输出是符合正态（高斯）分布的[⊖]，我们可以用t-检验去解答这个问题。$\mu_1=10.21$，$\sigma_1=1.112$，$\mu_2=8.85$，$\sigma_2=0.977$

用式3.6可得：

$$T=\frac{10.21-8.85}{\sqrt{(10-1)\ 1.112^2+(6-1)\ 0.997^2}}\sqrt{\frac{10\times 6\ (10+6-2)}{10+6}}=2.456$$

⊖ 在这种情况下，对这种假设而言，样本总数确实太少了，但是这种小样本可以使例子变得更清晰。

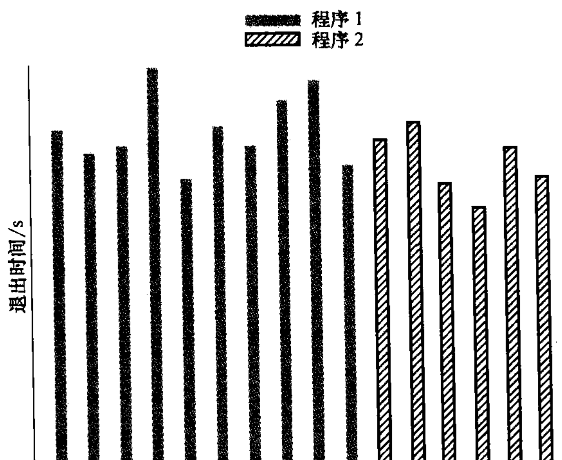


图 3.6　两个不同的从死角中退出的程序给出的结果

$k=10+6-2$，$t_{\alpha}=2.145$（由表3.3得到）。不等式 $|2.456|>2.145$ 成立，假设 H_0（$\mu_1=\mu_2$）被拒绝，这就说明第二个程序执行得比第一个好。这个说法错误的可能性是 0.05.

3.3.5　相关样本的 t-检验

如果进行的两个检验的在每一次检验中主体相同，例如相同的机器人执行任务 X 和任务 Y，从实验中获得的数据是配对的，或者说是“两者相关的”，这种情况下，下面的 t-检验就可以用了。

为进行检验，t_d统计量可由所给式（3.7）计算得到：

$$t_d=\frac{\mu_x-\mu_y}{S_D} \tag{3.7}$$

式中，μ_x 和 μ_y 是两个测量值的均值；S_D 是由式（3.8）给出的差别的标准误差。

$$S_D=\frac{1}{\sqrt{N}}\sqrt{\frac{\sum D^2-\frac{(\sum D)^2}{N}}{N-1}} \tag{3.8}$$

$D=X-Y$ 是任务 X 的得分与任务 y 相应的得分的差，N 是两个分数的和。这里的“得分”指的是用于估计的系统测量值。例如下例中，移动型机器人的速度。

为确定显著性，当 $k=N-1$ 自由度时，在理想的显著性水平下，比较由表 3.3 给出的 t_d 与临界值 t_{crit}。如果 $|t_d|>t_{crit}$，那么无效假设（两个样本无显著差异）被拒绝。

相关样本的 t-检验实例：橡胶轮与塑料轮

工厂运输机器人的轮子由硬塑料改成了软的橡胶轮，后者有更好的抓地力。把机器人放在不同表面的相同轨迹上运动。起初，我们给它们装上塑料轮，然后换成橡胶轮。每种轮子做 12 次实验，在每次实验中机器人的平均速度都被记录下来。我们希望知道用橡胶轮和塑料轮时机器人的速度是否存在显著差异。由于实验是在相同的环境下进行，用的是同一个机器人，所以这个实验是配对实验，样本也是相关的。

将获得的检验结果列在表 3.4 中，同时也列上一些对该检验有用的计算。

表 3.4 对同一个机器人，分别用橡胶轮和塑料轮执行相同的任务，得到的速度数据

													合计	μ
橡胶轮	58	31	82	87	40	42	71	45	35	23	13	24		45.9
塑料轮	59	146	111	111	75	60	57	81	13	59	106	37		76.3
D	-1	-115	-29	-24	-35	-18	14	-36	22	-36	-93	-13	-364	-30.3
D^2	1	13225	841	576	1225	324	196	1296	484	1296	8649	169	28282	

代入数据，式 3.8 $S_D=\frac{1}{\sqrt{12}}\sqrt{\frac{28282-\frac{(-364)^2}{12}}{12-1}}=11.4$. 再由 $\mu_{rubber}=45.9$ 和 $\mu_{plastic}=76.3$，式 3.7 得 $t_d=\frac{45.9-76.3}{11.4}=-2.66$

有 $k=N-1=11$ 个自由度，从表 3.3 中我们可以看出，显著性水平 5%，$t_{crit}\approx 2.2$ 。又 $|t_d|>t_{crit}$ 我们拒绝无效假设：用橡胶轮和塑料轮，获得的速度在统计意义上有显著差异。用橡胶轮的机器人更快。

在 Scilab 软件中，我们精确确定显著性水平用的是 [P, Q] = [P Q] = cdft ("PQ", T, Df)：

```
[P Q] = cdft ("PQ", 2.66, 11)
P =
    0.0110934
```

P 相当于在这个双尾检验中显著性水平的一半，所以结果的显著性是 $p=0.022$

另一种可以得到相同的结果的方法是计算一个接受区间。用 t 统计量确定问题中的样本统计量是否落在接受区间。

在这种情况下，我们的无效假设是：$H_0\text{: }\Delta_0=\mu_{rubber}-\mu_{plastic}=0$ 。速度的均值差是 $\mu_{diff}=30.3$ ，标准差 $\sigma_{diff}=39.6$ ，标准误差 $\overline{\sigma}_{diff}=\frac{\sigma_{diff}}{\sqrt{n}}=8.8$ 。在显著性水平为 5% 的双尾 t-检验中，我们可得到 t = 2.2 ，且表 3.3 中自由度 $k=12-1$ 。

因此接受区间是 $\Delta_0 \pm t \times \overline{\sigma}_{diff}$，如，在这个例子中，0 ±2.2×8.8（-19.3～19.3）。均值差30.3在接受区间外，因此速度的变化在5%的显著性水平上是显著的。

3.3.6 比较两个以上的样本：方差参数分析

在3.3.4中，我们讨论了参数数据的两种均值的显著差异是如何确定的。例如，我们运行移动型机器人让它们在两种不同表面的地面上工作，然后分别测量机器人的平均速度，想知道机器人的速度在两种情况下是否显著不同。

但是如果我们引入第三种情况，我们让机器人在第三种地面上运行，这时在3.3.4中介绍的t-检验就不再适用了。注意，不要尝试去反复比较三种情况中的两种！

然而，我们可以用方差参数分析（ANOVA），这种方法允许我们检验样本均值超过两个时差异的显著性，前提是假设总体分布都是（近似是）正态的（如果总体分布是非正态的，或者是未知的，用非参数的ANOVA检验来描述，参见3.3.4节内容）。

无效假设还是 $\mu_1 = \mu_2 \cdots = \mu_k$。ANOVA检验的总体假设是这样的：如果 k 样本是完全从相同的分布（无效假设）中取出的，那么我们应该能够用两种方法估计出总体方差：1）通过计算 k 样本均值的方差（行间变异数）；2）通过分别计算每个样本的方差（行内变异数）。如果所有的样本确实都来自同一个总体，不管我们是通过何种方法计算得到的，我们都会得到相同的方差。

为了进行方差分析，我们首先要确定行间变异数 σ_b^2（式3.9）和行内变异数 σ_w^2（式3.10）：

$$\sigma_b^2 = \frac{\sum n_j(\mu_j - \bar{\mu})^2}{k-1} \tag{3.9}$$

式中，n_j 是第 j 个样本的样本容量；μ_j 是第 j 个样本的均值；$\bar{\mu}$ 是所有相关样本的总均值；k 是样本的数量。

$$\sigma_w^2 = \sum \left(\frac{n_j - 1}{n_T - k}\right)\sigma_j^2 \tag{3.10}$$

式中，$n_T = \sum n_j$ 是总的样本容量；σ_j^2 是第 j 个样本的方差。

一旦我们确定了这两个方差，我们便可以由给出的式3.11计算出比例 F，进而检验它的显著性。

$$F = \sigma_b^2 / \sigma_w^2 \tag{3.11}$$

1. 参数的方差分析：显著性检验

为了检验显著性，我们用3.6.1节中表3.18给出的F-分布。分子自由度数

f_1 由等式 $f_1 = k - 1$ 给出，分母的自由度 f_2 由等式 $f_2 = n_T - k$ 给出。如果用式（3.11）计算出的 F 值超过了表3.18的临界值，那么无效假设就被拒绝，意味着在 k 个样本均值之间存在显著性差异。否则，无效假设 $\mu_1 = \mu_2 \cdots = \mu_k$ 就被接受。

2. 参数的方差分析举例："从死角中退出（Dead End Escape）"的再次讨论

在接下来的例子中我们将会重提在第3.3.4小节中的两个不同机器人控制程序"从死角中退出（Dead End Escape）"的例子。

为方便起见，每一个程序的"从死角中退出（Dead End Escape）"时间在表3.5中再次给出。

表3.5 两个不同"从死角中退出（Dead End Escape）"程序使用的时间

（单位：s）

程序 A	10.2	9.5	9.7	12.1	8.7	10.3	9.7	11.1	11.7	9.1
程序 B	9.6	10.1	8.2	7.5	9.3	8.4				

在这个例子中，我们可知 $n_A = 10$，$n_B = 10$，$n_T = 10$，$\bar{x}_A = 10.2$，$\sigma_A = 1.1$，$\bar{x}_B = 8.9$，$\sigma_B = 0.98$，$\bar{\mu} = 9.7$ 和 $k = 2$。

根据式（3.9）和式（3.10），我们可以得到

$$\sigma_b^2 = \frac{10\ (10.2 - 9.7)^2 + 6\ (8.9 - 9.7)^2}{2 - 1} = 6.3$$

$\sigma_w^2 = \frac{10 - 1}{16 - 2} 1.1^2 + \frac{6 - 1}{16 - 2} 0.98^2 = 1.1$，由这些数据可得 F 值 $F = \frac{6.3}{1.1} = 5.7$。

统计学分析的自由度是 $f_1 = k - 1 = 1$ 和 $f_2 = n_T - k = 14$。从表3.18我们看到临界值 $F_{crit} = 4.60$。计算的 F 值超出了临界值 F_{crit}，因此我们拒绝无效假设，进而确定在两个程序的性能之间存在显著的差异。

3.4 样本比较的非参数方法

有些实验不产生实际的测量值（比如速度、时间等），只是性能的等级。例如，两个地面清洁机器人可能受到人类志愿者的评估，即观察机器人的清扫过程和评估在任意范围内它们运行的质量。实验者或许想知道这两个机器人在性能上实质是否相同，或者其中一个机器人感知得比另一个要好。参数法比如t-检验不能用于这种情况，因为没有参数数据是可用的。还有一种情况就是被分析的数据不是正态分布的，但是服从另外的某一种分布。这就是非参数法涉及的领域。

与参数法不同，非参数法对要分析的数据分布不做任何假设，因此显得"更强大"因为在任何数据分布下都能一样分析。尽管如此，它们其实还是比参数法（可以拒绝无效假设）要弱一点。还有，在参数分析法也能使用的时候，就要考虑两种方法的选择问题。

多数情况下非参数分析法分析等级，例如，在一个特别的任务中取得的分数有多高或有多低，然后比较等级分布是否符合特定的无效假设。在这一节中我们会看到基于等级的非参数法。

3.4.1 中值以及中值的置信区间

在正态分布的情形下，我们利用式（3.2）已经解出了“期望值”、均值，通过式（3.4）又可以得出均值的置信区间。

对于非正态分布的数据或者对于未知其分布规律的数据，一个相似的方法可以考虑中值。中值仅仅是我们数据里的中间值。例如，如果从一个噪声传感器获得测量数据［18 22 25 29 43 59 67 88 89］，那么43就是中间值，因此也是中值（对于有偶数组的数据中值由中间两个值的平均值决定）。

为了解出中值的置信区间，我们可以利用表3.6。此表显示了当 r 的值从数据集中极远的两点向中间靠拢时，置信区间的上下边界。

表3.6 中值的置信区间，当 r 的值从数据集中极远的两点向中间靠拢时

（5%的显著性水平，参考［Barnard et al，1993］）

n	1～5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
$r_{5\%}$	无效	1	1	1	2	2	2	3	3	3	4	4	5	5	5	6	6	6	7	7	8	8	8	9	9	10

在我们的例子中，$n=9$，我们可以得到 $r=2$，在5%的显著性水平下中值的置信区间是［22，88］。由于在这种情形下存在均值和标准误差的缘故，中值经常和它的置信区间一起分析，例如在条形图中，或者引用文献时。

3.4.2 非参数检验对于两个分布的差别（曼-惠特尼 U-检验或是威尔科克森秩和检验）

曼-惠特尼（Mann-Whitney）U-检验，又称为威尔科克森（Wilcoxon）秩和检验，决定两个不同的样本[1]是否从相同或者不同的两个总体中抽取。它可以看作对独立样本进行t-检验的非参数法，还可以用来比较有不同或相同样本容量的样本。

检验的无效假设是中值和分布的形状都是相同的，备择假设[2]则不是这样。因为 U-检验不需要任何总体分布的参数的信息，它对小样本（那些不能确定它们是否是正态分布的样本）或者被认为是非正态分布的样本特别有用。U-检验的进一步作用是它可以应用到任意测量得到的数据，只要这些数据是有序的：U-检验是一种非参数检验，没有对总体分布作出任何假设。

---

[1] 曼-惠特尼（Mann-Whitney）U-检验只能用于比较两个不同的分布。要想比较两个以上的数据集，就会用到非参数法ANOVA（第3.4.4节）。

[2] 假设有零假设（H_0）和备择假设（H_1）两个，零假设又称无效假设或检验假设。H_0 和 H_1 的关系是互相对立的，如果拒绝 H_0，就要接受 H_1。——译者注。

1. 对差别的非参数检验：例 1

例如，我们假定一个特殊的家用机器人在日本和欧洲同时上市。制造商想要知道这两个市场的客户满意度是否有显著差异（在 5% 的显著性水平），因此获得了来自欧洲的 15 个和日本的 14 个家庭的客户满意度评分（0～100 分）。

获得的性能评分在表 3.7 中列出。

表 3.7　两个不同市场上该机器人获得的客户满意度评分

欧洲	85	60	90	10	33	67	70	76	33	89	95	50	15	45	56
日本	60	65	30	25	45	70	50	43	35	61	70	30	29	56	

表 3.7 列出的数据是非正态分布的，曼-惠特尼 U-检验可用于研究两个市场获得的得分没有显著差异的无效假设。

为了进行这一检验，首先应该给这些评分数据列出等级，就像在表 3.8 中列出的那样（注意对结合秩，秩中间插入了平均值）。

表 3.8　在欧洲和日本获得的等级分数

等级	分数	市场	欧洲分数	日本分数
1	95	欧洲	1	
2	90	欧洲	2	
3	89	欧洲	3	
4	85	欧洲	4	
5	76	欧洲	5	
7	70	欧洲	7	
7	70	日本		7
7	70	日本		7
9	67	欧洲	9	
10	65	日本		10
11	61	日本		11
12.5	60	欧洲	12.5	
12.5	60	日本		12.5
14.5	56	欧洲	14.5	
14.5	56	日本		14.5
16.5	50	欧洲	16.5	
16.5	50	日本		16.5
18.5	45	欧洲	18.5	
18.5	45	日本		18.5
20	43	日本		20

（续）

等级	分数	市场	欧洲分数	日本分数
21	43	日本		21
22.5	35	欧洲	22.5	
22.5	33	欧洲	22.5	
24.5	33	日本		24.5
24.5	30	日本		24.5
26	29	日本		26
27	25	日本		27
28	15	欧洲	28	
29	10	欧洲	29	
总计			195	240

接下来计算每一个样本的所谓 U-统计量，U-统计量的均值（式 3.13）和 U-统计量的标准误差（式 3.14），其中 n_1 和 n_2 分别是组 1 和组 2 中得分的个数，R_1 和 R_2 分别是组 1 和组 2 中秩的和。

$$U_1 = n_1 n_2 + \frac{n_1 (n_1 + 1)}{2} - R_1 \tag{3.12}$$

$$U_2 = n_1 n_2 + \frac{n_2 (n_2 + 1)}{2} - R_2$$

$$\mu_u = \frac{n_1 n_2}{2} \tag{3.13}$$

$$\sigma_u = \sqrt{\frac{n_1 n_2 (n_1 + n_2 + 1)}{12}} \tag{3.14}$$

在这个例子中，得到下面的结果

$$\begin{aligned} U_1 &= 15 \times 14 + \frac{15 (15 + 1)}{2} - 195 = 135 \\ U_2 &= 15 \times 14 + \frac{14 (14 + 1)}{2} - 240 = 75 \\ \mu_u &= \frac{15 \times 14}{2} = 105 \\ \sigma_u &= \sqrt{\frac{15 \times 14 (15 + 14 + 1)}{12}} = 22.9 \end{aligned} \tag{3.15}$$

如果 n_1 和 n_2 均大于 10，那么 U-统计量近似于正态分布［Levin and Rubin, 1980, p. 486］，这里的例子就是这种情况。对于小样本空间，可以参考后面介绍的例 2。

图 3.7 显示了这种情况。我们比较感兴趣于确定这两个分布的不同是否在 5% 的显著性水平上，例如，U 值是否显著地高于或者低于均值，这因此是一个双尾检验。我们感兴趣于确定 U 是否位于接受域，或者在接受域外面 5% 的地方。

或许确定这一点的最简单方式是利用一个标准的正态分布表，比如说表 3.2。最想确定的是 z 值，正态分布曲线在 z 和 μ 下方区域等于在图 3.7 中显示的接受域的大小。这种情况下，这个区域应该占据了正态分布曲线下方的 47.5% 的区域，因为在 5% 的时域得到的显著性，用的是双尾检验。

从表 3.2 可以看出，对于 $z=1.96$，在 μ 和 z 之间在正态分布曲线下方的面积是 0.475（47.5% 的面积在正态分布曲线下方），因此 $z=1.96$ 就是在这个例子中要寻找的答案。

接受域的上下限由式（3.16）给出

$$\mu_u - z\sigma_u < A < \mu_u + z\sigma_u \tag{3.16}$$

式中，A 是接受域；μ_u 是在式（3.13）中给出的 U-统计量的均值，σ_u 是由式（3.14）给出的 U-统计量的标准误差。

在这种情况下我们可以得到 $105-1.96\times22.9<A<105+1.96\times22.9$，这就得到了一个从 60.12 到 149.88 的接受域（见图 3.7）。

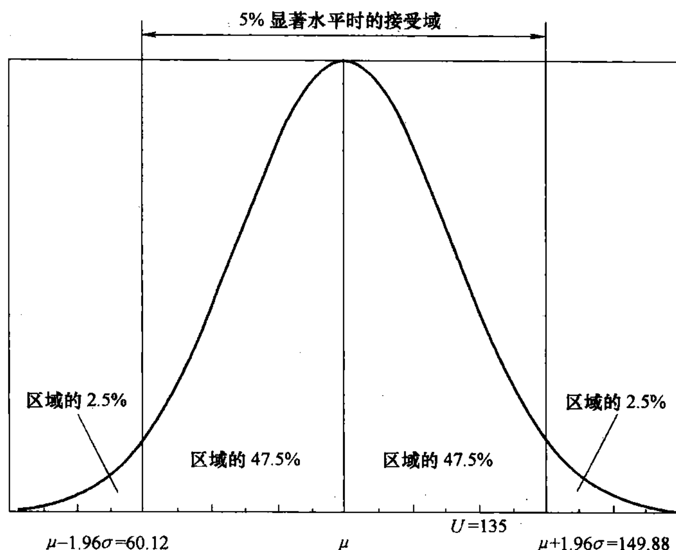


图 3.7 家用机器人例子的图形表示

$U_1=135$ 与 $U_2=75$，显然在接受域内，我们因此没有理由拒绝显著性水平为 5% 时的无效假设（因此，我们可以假定两个分布中，机器人的等级没有显著性差异）。

看一下表3.2，我们可以进一步看到41%的面积在正态分布曲线下面（例如，18%的显著性水平）我们可以得到$z=1.34$，导出了接受域是$105 \pm 1.34 \times 22.9$（74.3~135.7）。这意味着即使在18%的显著性水平上，我们依然不能拒绝在两个市场上机器人的客户满意度的等级同样高的无效假设。

z值当然也能用软件包轻易地解出。在Scilab软件中求z时通过输入：

```
Cdfnor ("X", 0, 1, P, Q)
```

Q是所要求的显著性水平的一半，P = 1 − Q。在这种特别的情形下就是

```
Cdfnor ("X", 0, 1, 0.975, 0.025)
    Ans  =
         1.959964
```

同样，对于某一特定的值U的显著性水平可以通过公式在Scilab中确定，公式如下：

```
[Q  P] = cdfnor ("PQ", U , mu , sigm a )
```

式中，P是显著性水平的一半（双尾检验!）；*mu*是U的统计量（式3.13）；*sigma*是U统计量的标准误差（式3.14）。

2. *U*检验例2：从死角中退出（Dead End Escape）重现

在3.3.4节中我们通过两个不同的控制程序分析了从死角中退出（Dead End Escape）的行为。为了方便，我们把程序A和程序B所得的结果都再次列在表3.9里。

表3.9 通过两个不同的从死角中退出（Dead End Escape）程序获得的数据结果

程序A	10.2	9.5	9.7	12.1	8.7	10.3	9.7	11.1	11.7	9.1	和
A的秩	5	10	7.5	1	13	4	7.5	3	2	12	65
程序B	9.6	10.1	8.2	7.5	9.3	8.4					
B的秩	9	6	15	16	11	14					71

在这里，样本的容量小于10，不是假定的U统计量的正态分布，而是来比较一下在U-检验中两个U_1和U_2值对临界值U_{crit}的影响，数据见表3.10。如果U_1和U_2中较小的数小于临界值U_{crit}，我们就拒绝无效假设。

表3.10 在样本容量n小于10，5%的显著性水平时U检验的临界值

	n_1																	
	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$n_2=3$	—	0	0	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8
$n_2=4$	—	0	1	2	3	4	4	5	6	7	8	9	10	11	11	12	13	14
$n_2=5$	0	1	2	3	5	6	7	8	9	11	12	13	14	15	17	18	19	20

（续）

	n_1																	
	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$n_2=6$	1	2	3	5	6	8	10	11	13	14	16	17	19	21	22	24	25	27
$n_2=7$	1	3	5	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34
$n_2=8$	2	4	6	8	10	13	15	17	19	22	24	26	29	31	34	36	38	41
$n_2=9$	2	4	7	10	12	15	17	20	23	26	28	31	34	37	39	42	45	48
$n_2=10$	3	5	8	11	14	17	20	23	26	29	33	36	39	42	45	48	52	55

我们从式（3.12）中得到：

$$U_1 = 10 \times 6 + \frac{10(10+1)}{2} - 65 = 50 \tag{3.17}$$

$$U_2 = 10 \times 6 + \frac{6(6+1)}{2} - 71 = 10 \tag{3.18}$$

由于 $U_2 < U_{crit} = 11$（见表 3.10），我们拒绝无效假设，确认了我们早期的结论：在两个不同的程序性能上有着显著的差异。

3.4.3 配对观察值的威尔科克森（Wilcoxor）检验

这是在 3.4.2 节所介绍检验的一种变化，它可以用于以某些标准来看实验结果是配对的实验。

1. 配对观察值：例 1

让我们假定一个机器人制造商生产两种型号的家用机器人，并且我们想知道主要市场的客户更喜欢其中哪一种模型的。因此，选择了 15 个潜在顾客，要求他们就满意度，对模型 1（M1）和模型 2（M2）的机器人给以从 0 到 100 分的主观评价。在这种情况下分数是配对出现的，因为每个客户都要评价这两种机器人。得到的结果如表 3.11 中的第一和第二行所示。

表 3.11　以百分制主观评价两种模型的机器人（M1 M2）的满意度调查

1	M1	54	62	67	42	13	89	56	45	68	23	30	24	35	87	70	
2	M2	54	38	43	8	22	73	50	48	27	26	13	27	67	66	66	
3	M1 - M2	0	24	24	34	-9	16	6	-3	41	-3	17	-3	-32	21	4	
4	秩		10.5	10.5	13	6	7	5	2	14	2	8	2	12	9	4	
																	和
5	秩 +		10.5	10.5	13		7	5		14		8			9	4	$T_+=81$
6	秩 -					6			2		2		2	12			$T_-=24$

无效假设是客户对两种模型的机器人的评价是没有显著差异的，换句话说，

就是两者之间的满意度差异与零没有显著性差异。制造商想在10%的显著性水平上来检验这一假设。

2. 检验假设

要进行检验，我们首先需要计算在评价时每对数值的差。如果无效假设确实是正确的，那么模型1获得高分的概率和模型2获得高分的概率是相等的。这个检验就是基于这个假设进行的。

计算出这些结果的差（表3.11的第三行）之后，我们对这些差的绝对值取秩，方法和*U*-检验（3.4.2节）中的取秩类似。差为零的可以忽略不计，如前所述，这些结合秩是所有结合值的平均秩。表3.11的第四行就是计算的结果。

然后，我们计算所有大于零的秩的和T_+和所有小于零的秩的和T_-（表3.11中的第五行和第六行）。

期望值μ_T和标准差σ_T由式（3.19）和式（3.20）给出：

$$\mu_T = \frac{n(n+1)}{4} \tag{3.19}$$

$$\sigma_T = \sqrt{\frac{(2n+1)\ \mu_T}{6}} \tag{3.20}$$

式中，n为差值为非零的个数。

对于下面这个特殊的例子，我们可以得到：

$$\mu_T = \frac{14(14+1)}{4} = 52.5$$

$$\sigma_T = \sqrt{\frac{(2\times14+1)52.5}{6}} = 15.9$$

威尔科克森检验表明如果差值中的非零个数n大于8（$n=14$就是这种情况），那么无效假设就被接受，这时T_+和T_-大约服从一个正态分布［Wilcoxon, 1947］。因此，我们可以套用一个与3.4.2节相似的技巧，把T_+和T_-与标准正态分布比较，如表3.1所示。如果T_+和T_-在式（3.16）给出的接受域以外，无效假设被拒绝，否则无效假设被接受。

对于10%的显著性水平，我们从表3.1中找到$z=1.65$，表3.1给出了T_+和T_-的接受域：$52.5\pm1.65\times15.9$（也就是$26.3\sim78.7$）。T_+和T_-在接受域以外时，我们认为在10%的显著性水平时，无效假设被拒绝：两个机器人的顾客满意度不是恒等的。

或者，换句话说，给定$z=1.79$，我们可以得到接受域为$52.5\pm1.79\times15.9$（也就是$24\sim80.9$）。通过双尾检验我们可以得到$z=1.79$所对应的显著性水平是7.4%，意味着在7.4%的显著性水平时，我们将拒绝无效假设。

在 Scilab 中显著性水平被定义为：

[Q P] = cdfnor("PQ ", TP, *mu*, *sigma*)

P = 0.0367970

其中，TP 是 T_+，mu = μ_T，sigma = σ_T，P 为显著性水平的一半（双尾检验）。

配对观察值 例2：橡胶轮子和塑料轮子再现

让我们看一下之前对这个例子的分析，对装有橡胶轮胎和塑料轮胎的机器人速度的比较（3.3.5 节）。

机器人可以达到的速度如表 3.12 所示。

表 3.12 同一个机器人用橡胶轮子和塑料轮子各自所达到的速度，每组数据都是执行相同的任务所得到的

橡胶轮子	58	31	82	87	40	42	71	45	35	23	13	24	
塑料轮子	59	146	111	111	75	60	57	81	13	59	106	37	
差值	1	115	29	24	35	18	-14	36	-22	36	93	13	和
秩	1	12	7	6	8	4	3	9.5	5	9.5	11	2	
秩 +	1	12	7	6	8	4		9.5		9.5	11	2	$T_+ = 70$
秩 -							3		5				$T_- = 8$

根据式（3.19）和式（3.20）我们可以得到

$$\mu_T = \frac{12(12+1)}{4} = 39$$

$$\sigma_T = \sqrt{\frac{(2\times 12+1)\times 39}{6}} = 12.7$$

我们再次给定一个 5% 的显著性水平，我们从表 3.2 中可以找到 $z = 1.96$，这样就可以确定接受域为 $39 \pm 1.96 \times 12.7 = [14, 64]$。这时 T_+ 和 T_- 在接受区间以外，因此拒绝无效假设：橡胶轮和塑料轮所能达到的速度有显著性差异。

3. 配对观察值 例2：路径规划比较

在相同的环境下，比较同一个机器人在两种不同的规划路径上运动的情形。得到的结果如表 3.13 所示。

同样，我们计算机器人在性能上的差异，对它们的绝对值取秩列在表 3.14 中。差为零的忽略不计，结合秩取决于它们的平均秩。

表 3.13 两种路径的性能指标，同样的机器人执行同样的任务

A	30	50	40	70	20	50	40	80	70	10	10
B	20	70	30	60	50	50	50	60	20	50	20

表 3.14 对表 3.13 的数据进行非参数分析

性能 A	30	50	40	70	20	50	40	80	70	10	10	和
性能 B	20	70	30	60	50	50	50	60	20	50	20	

（续）

差值	10	-20	10	10	-30	0	-10	20	50	-40	-10	
秩$_A$	3		3	3		—		6.5	10			$T_A=25.5$
秩$_B$		6.5			8	—	3			9	3	$T_B=29.5$

由式（3.19）和式（3.20），我们可以得到：

$$\mu_T=\frac{10(10+1)}{4}=27.5$$

$$\sigma_T=\sqrt{\frac{(2\times10+1)\quad\times27.5}{6}}=9.81$$

在这个例子中，在8%的显著性水平下，我们很希望可以确定这两种规划路径是否有显著的差异。从3.2节中我们可以得到在这个显著性水平下 $z=1.75$；因此接受区间是 $27.5\pm1.75\times9.81$（也就是10.3～44.7，根据式3.16）。T_A 和 T_B 都在接受区间内，因此无效假设（两种规划路径没有显著差异）不可以被拒绝。

3.4.4 两组或更多组的差别检验（非参数方差分析，克鲁斯卡尔-沃利斯检验）

在3.3.6节给出了一种样本多于两个时，确定它们是否有显著性差异的方法，假设总体分布是正态的。

然而，这个假设并不总是满足的，本节所描述的非参数方差分析可用于在以上这些情况下。正如在其他非参数检验中，非参数方差分析是基于秩（等级）的。

正如在3.3.6节中所讨论的参数方差分析，我们很希望可以确定 k 组数据有无显著性差异。

要进行检验，就要从所有组中合并数据，然后对它取秩（通常，结合秩取决于平均值）。然后，把每组得到的秩相加，得到 R_k 与 k。

然后，我们由所给的式（3.21）计算 H 的统计量：

$$H=\frac{12}{N(N+1)}\left(\sum_{i=1}^{k}\frac{R_i^2}{n_i}\right)-3(N+1) \tag{3.21}$$

式中，N 是所有 k 组点的数量总和；n_i 是第 i 组点的数量。

如果每个 k 样本包含至少五个观测值，那么 H 统计量的样本分布十分接近于 χ^2 分布（甚至三个观测值都很接近）。因此，要想确定所有组之间有无差异，对于 $k-1$ 个自由度，我们就要将 H 统计量和给定的 χ^2 表（表3.24）中的临界值相比较。如果 H 超出了表中所给的临界值，那么拒绝无效假设（每组之间没有显著的差异）。

3.4.5 克鲁斯卡尔-沃利斯检验举例：从死角中退出（Dead End Escape）重现

在前面，我们已经确定了两个不同的从死角中退出（Dead End Escape）程序有无显著的差异。我们已经用过了 t-检验（3.3.4 节）和曼-惠特尼 U-检验（3.4.2 节）。现在我们将用第三种检验方法——克鲁斯卡尔-沃利斯（Kruskal Wallis）来分析表 3.15 中的数据。

表 3.15 两种不同的从死角中退出（Dead End Escape）程序获得的结果

程序 A	10.2	9.5	9.7	12.1	8.7	10.3	9.7	11.1	11.7	9.1	和
矩阵的秩 A	5	10	7.5	1	13	4	7.5	3	2	12	65
程序 B	9.6	10.1	8.2	7.5	9.3	8.4					
矩阵的秩 B	9	6	15	16	11	14					71

在这个特殊的例子中，我们知道 $k=2$，$N=16$。根据式（3.21）我们可以计算 H：

$$H=\frac{12}{16\times(16+1)}\left(\frac{65^2}{10}+\frac{71^2}{6}\right)-3\times(16+1)=4.7$$

对于自由度 $k-1=1$，可以从后面介绍的表 3.24 中得到临界值 $\chi_{0.05}=3.84$。H 超过了临界值 $\chi_{0.05}$，因此无效假设被拒绝，原来确定的两个从死角中退出（Dead End Escape）行为有显著性差异的结论是正确的。

3.5 序列的随机性检验

有时候知道两个样本（例如一项检验的结果、一些运算的结果等）是否以随机的次序出现，或两个样本被取出时是否按照一定的次序是一件有趣的事。例如，我们需要这个信息来决定一个时间序列是否是稳定的（参看 4.3.2 节）。

通过举例，假设拥有一台倾向于以随机的方式移动的保洁机器人。记录下机器人向左和右转的序列，想知道他们是否真的是随机分布的。得到了如下的序列：左左右右左左右右左左右右左左右右左左左右右右左左左右右右左左左左右右右右左左左左右右右右左左右右。这个序列看起来一点也不像随机的，而接下来的一个样本游程检验 [Bendat and Piesol, 2000] 将会证实这个序列确实不是随机的。

需要确定标志 1（左）和标志 2（右）出现的频数 n_1 和 n_2，以及游程个数 γ，即相同标志的子序列，来进行检验。在这个事件中 $n_1=n_2=24$，$\gamma=18$（左左-右右-左左-右右-左左-右右-左左-右右-左左左-右右右-左左左-右右右-左左左

左-右右右右-左左左左-右右右右-左左-右右）。

均值 μ_γ 和统计量 γ 的标准误差 σ_γ 在式（3.22）和式（3.23）中给出。在此例中得到 $\mu_\gamma=25$，$\sigma_\gamma=3.4$.

$$\mu_\gamma = \frac{2n_1n_2}{n_1+n_1}+1 \tag{3.22}$$

$$\sigma_\gamma = \sqrt{\frac{2n_1n_2(2n_1n_2-n_1-n_2)}{(n_1+n_2)^2(n_1+n_2-1)}} \tag{3.23}$$

式中，n_1 是标记 1 的出现频数；n_2 是标记 2 的出现频数。

如果 n_1 或 n_2 大于 20，则下面的游程检验可以应用，因为这个事件中游程 γ 的分布服从正态分布，这意味着我们可以使用我们常规步骤确定一个置信区间，如果我们的无效假设（H0：标记以随机的次序出现）成立，那么实际观测的组数 γ 会落于其中。以显著性水平为 0.05 来做一次分析；图 3.8 展示出了这种情况。

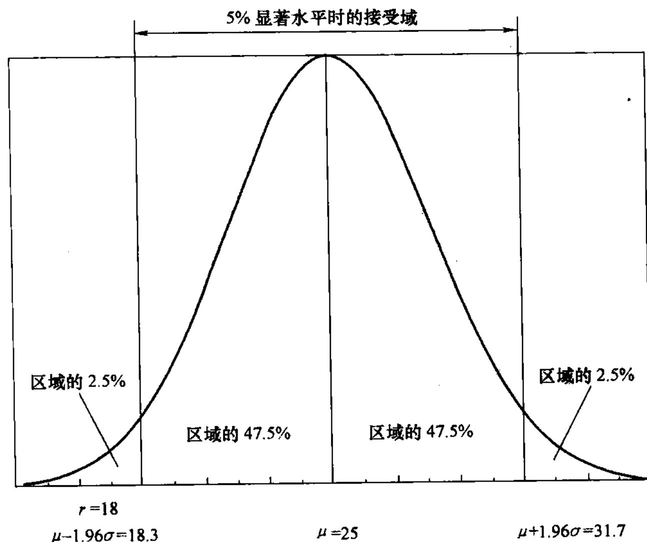


图 3.8　3.5 节所给例子的接受区间

此检验意图确定我们的观测游程个数 γ 是否在 $\mu_\gamma \pm z\sigma_\gamma$ 的接受区间内，即在接受下限以上，接受上限以下。因此，这是一个双尾检验，在正态分布曲线以下的接受区间占了此曲线下部 95% 的区域。

从表 3.2 中我们发现当显著性水平为 0.05（即在 μ_γ 和大小为 0.475 的上置信区间的末端之间的区域）$z=1.96$，接受区间因此为 $25\pm1.96\times3.4=[18.3, 31.7]$。$\gamma=18$ 在接受区间之外，我们因此拒绝无效假设：机器人的转弯不是按随机的序列进行的。

下面的 Scilab 代码将会对由两个字符“T”和“F”代表的序列的随机性进

行一个游程检验。

```
function [ ] = runstest(run)
// Ulrich Nehmzow
// 确定一个序列的“游程”是否是随机的
// “游程”是真和假的比例(个数)的序列

// 计算 n1,n2 和 r
n1 = sum(run == %T)
n2 = sum(run == %F)
r = 1

if((n1 < 20)&(n2 < 20))
        printf ("There is not enough data to conduct the test \n")
        abort
end
last = run(1)
for i = 2:length(run)
        if(last ~= run(i))
                r = r + 1
                last = run(i)
        end
end
// 现在对随机性进行检验
mur = 1 + (2 * n1 * n2)/(n1 + n2)
sigmar = sqrt((2 * n1 * n2 * (2 * n1 * n2 - n1 - n2))/((n1 + n2 - 1) * (n1 + n2)^2))
// [P Q] = cdfnor("PQ",r,mur,sigmar)// P/2 是显著性水平

P = input("Please enter the desired significance level')
lowerbound = cdfnor("X",mur,sigmar,P/2,1 - P/2)
upperbound = cdfnor("X",mur,sigmar,1 - P/2,P/2)
printf("The accept. interval for r is %4.1f to %4.1f \n",lowerbound,upperbound)
if((r > lowerbound)&(r < upperbound))
        printf("r = %d is within that region,therefore the number of runs
        is random H0 is accepted. \n",r)
else
        printf("r = %d is outside that region,therefore the number of runs is not
        random,H0 is rejected. \n",r)
end
```

3.6 趋势的参数检验（相关分析）

前面的部分解决了确定两组（或更多）数据是否有统计上显著差异这样一个问题。它们没有解决两组数据是否相关的问题，即第一个数据集是否对第二个数据集有所揭示，换言之，在两个变量间是否存在一种因果关系。

因此在这一节里，我们将着眼于一种检验，它可以精确确立：一个数据集是否被另一个数据集所预测（如果是，那么程度如何），或有没有两个相关的数据集？

在趋势检验里选择了两个数据集之一，X 作为自变量，并确定另一个数据集，Y 对 X 的依赖度如何，Y 则叫做因变量。

3.6.1 参数线性回归分析

到目前为止，所讨论的检验并没有研究两组数间是否存在一种因果关系。然而，在许多实验情况中，这种关系是存在的。例如，人们的体重和身高间存在一种因果关系（它们是“相关的”）。如果选择两个变量之一作为自变量（即被用户选择），则令一个变量取决于这一选择：如果确定一个人的身高为 1.8m，则他的体重将会在一些置信区间里被“决定”。

回归分析研究了自变量和因变量之间的关系。它是一种参数检验，并假定因变量是正态分布的（这对于自变量来说是非必须的）。

例如，假设测量了一个移动机器人的速度和电池电量，得到了表 3.16 中所示数值。将二者绘制成图，这些数值反映在图 3.9 中的散点图上。

以肉眼来看，电池电量和机器人速度之间显然存在一定的因果关系，这从数据连线中更加明显。我们希望通过数学方法了解到这种关系是什么，并且它是否是显著性的。

表 3.16 例：电池电量与机器人速度的关系

电量/V	12.0	11.7	11.5	11.1	10.8	10.4
速度/（cm/s）	41	45	41	40	32	34

1. 线性回归

给定自变量（电池电量），运用图 3.9 的线性关系我们可以预测因变量（机器人速度）。线性回归是确定式（3.24）的直线方程中参数 a 和 b 的途径：

$$Y = aX + b \tag{3.24}$$

当 Y 是因变量，X 为自变量时，斜率 a 和截距 b 的定义如式（3.25）和式（3.26）：

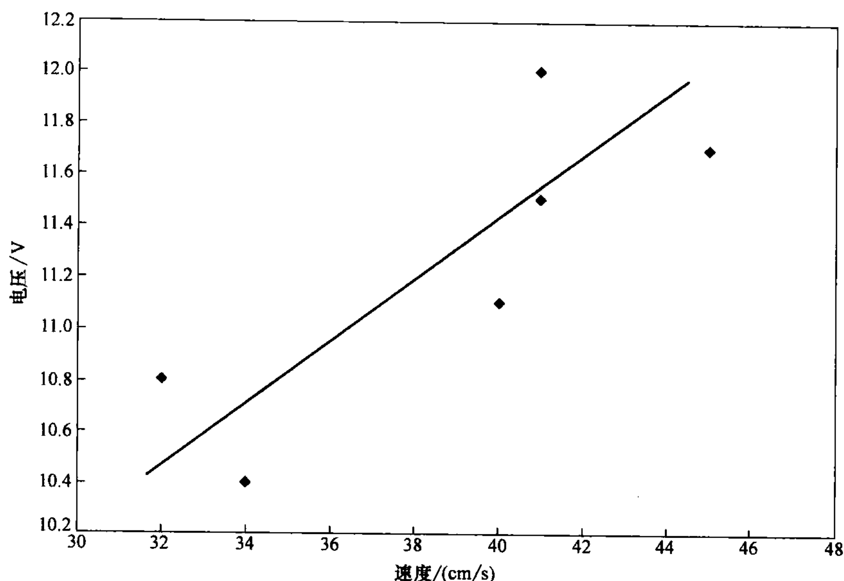


图 3.9 例：电池电压与移动机器人速度的关系

$$b = \frac{\sum XY - n\overline{X}\,\overline{Y}}{\sum X^2 - n\,\overline{X}^2} \tag{3.25}$$

$$a = \overline{Y} - b\,\overline{X} \tag{3.26}$$

式中，X，Y 分别为自变量和因变量；$\overline{X}$，$\overline{Y}$ 是它们的均值；n 是相关的自变量和因变量的对数。

在这个例子中自变量（电池电压）如表 3.16 所示。由之后的式（3.25）和式（3.26）可以得到如式（3.27）中所给出的电压（V）和速度（v）的线性关系：

$$v = 6.7Q - 36.9 \tag{3.27}$$

除了用式（3.25）和式（3.26）计算 a 和 b，它们还可以用数学软件包来求得。在 Scilab 中用以下的程序可以计算。

```
-→ [a  b] =reglin (charge, speed)
b =
-36.906103
a =
6.7323944
```

这是给定电池电压后对机器人速度的一个预测，而接下来的问题是：这是一个好的预测吗？

式（3.28）中定义的估计量 s_e 的标准误差解决了这个问题。它测量回归线

周围的被观测数据的散点：

$$s_e = \sqrt{\frac{\sum(Y-\hat{Y})^2}{n-2}} = \sqrt{\frac{\sum Y^2 - a\sum Y - b\sum XY}{n-2}} \tag{3.28}$$

式中，X 为自变量值；Y 为一个因变量值；$\hat{Y}$为一个通过式（3.24）估计后的因变量值；n 为用以构成回归直线的自变量和因变量的配对数。

估计标准误差越大，回归直线周围的散点越偏离直线，则自变量和因变量之间的相关性越弱。从另一方面说，当 $s_e=0$，我们将在给定的自变量下得到一个对因变量的完美预期。

在下面给出的特例中，我们得到其估计量的标准误差 $s_e=\sqrt{\frac{39.15}{6-2}}=3.13$，如表 3.17 所示。

同之前定义的标准差类似，假设观测点在回归线周围正态分布，我们期望 68% 的点落在回归线周围 $\pm 1s_e$ 的区间内，95.5% 的点在回归线周围 $\pm 2s_e$ 的区间内，99.7% 的点在回归线周围 $\pm 3s_e$的区间内。

表 3.17 计算表 3.16 所示例子中的标准误差估计

X	Y	$\hat{Y}=6.7X-36.9$	$(Y-\hat{Y})^2$
12	41	43.5	6.25
11.7	45	41.5	12.32
11.5	41	40.2	0.72
11.1	40	37.5	6.40
10.8	32	35.5	11.97
10.4	34	32.8	1.49
			$\sum 39.15$

这让我们可以对我们所做预测中的置信情况给以说明，例如，如果我们想预测机器人的速度，给定一个 11.3V 的电池电压，我们预期（式 3.27 ）$v=6.7Q-36.9=38.8$ 。现在我们可以说有 68% 的把握确定机器人的速度将会在区间 38.8 ± 3.13 这个区间里，或 95.5% 的把握确定它会在 $38.8\pm2\times3.13$ 的区间内。

严格地说，这些计算只适用于样本大小 $n>30$，因为对于更小的样本容量用正态分布的预测区间是不正确的，因此得到的结论就会是不准确的。然而，它们说明了原理。要对更小样本容量适用，你需要用到表 3.3 中所示的 t-分布。例如，如果我们想要有 95% 的把握确定机器人的真实速度落在所计算的置信区间中，在 $k=n-2=4$ 的自由度和 5% 显著性水平下根据表 3.3 可查得 $t=2.776$。因此，机器人的速度有 95% 的确定性落于区间 $38.8\pm2.776\times3.13$ 中。

最后请注意自变量和因变量之间的非线性关系。尽管线性回归分析假设的

是一种线性关系，而且基于此假设检验了显著性，通过变换数据集可建立一种线性关系，这同样适用于非线性关系。例如，对数变换会将自变量和因变量之间的指数关系线性化。

2. 线性回归：显著性检验（F-统计量）

有两种可以用于对线性回归显著性的检验：1）检验自变量方差的相关改变是否是变量方差改变的原因（F-检验）；2）检验回归线和数据点（“错误”的）间的差异是否显著地不同于零（t-检验）。这两个检验下面都将讨论——如果对任何一个错误，在自变量和因变量间没有显著的因果关系，那么最好不要用回归线作预测。

（1）F-统计量　F-统计量允许我们检验显著性以及因变量 Y 的方差是否因自变量 X 的方差的变化而改变。F 由式（3.29）定义如下：

$$F = \frac{RSS}{DMS} \tag{3.29}$$

$$RSS = \frac{S_{XY}^2}{S_{XX}} \tag{3.30}$$

$$DMS = \frac{DSS}{n-2} \tag{3.31}$$

$$DSS = S_{YY} - \frac{S_{XY}^2}{S_{XX}} \tag{3.32}$$

$$S_{XX} = \sum X^2 - \frac{(\sum X)^2}{n} \tag{3.33}$$

$$S_{YY} = \sum Y^2 - \frac{(\sum Y)^2}{n} \tag{3.34}$$

$$S_{XY} = \sum XY - \frac{(\sum X)(\sum Y)}{n} \tag{3.35}$$

式中，n 为用于回归分析的数据对的数量。

将结果的 F 值与表3.18所示临界值相比较，分子自由度是 $f_1=1$，分母自由度为 $f_2=n-2$。表3.18给出了5%显著性水平的临界值。显然计算得到的 F 值肯定比表3.18中所示的那些值大。关于其他显著性水平的数值可以在统计表中找到或计算得到。在Scilab中，它是由以下的命令得到的：

```
F = cdff("F",DFN,DFD,Q,P)
```

其中，DFN是分子的自由度，DFD分母的自由度，P是显著性水平，Q=1-P。

回到表3.16所示例子中，我们得到 $S_{YY}=118.83$，$S_{XY}=11.95$ 。从式（3.29）得到 $F=8.38$ 。F 在 $f_1=1$，$f_2=6-2=4$ 时的临界值是 $F_{crit}=7.7$（见表3.18）。则计算所得的 F 值8.38比它大，这表明电池电压和机器人速度间的相关性在5%的水平时是显著的。

表 3.18 F 分布（$p=0.05$）临界值（f_1 值是分子自由度，f_2 值是分母自由度）

f_1 \ f_2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	∞
1	161.45	18.51	10.13	7.71	6.61	5.99	5.59	5.32	5.12	4.96	4.84	4.75	4.67	4.60	4.54	4.49	4.45	4.41	4.38	4.35	3.84
2	199.50	19.00	9.55	6.94	5.79	5.14	4.74	4.46	4.26	4.10	3.98	3.89	3.81	3.74	3.68	3.63	3.59	3.55	3.52	3.49	3.00
3	215.71	19.16	9.28	6.59	5.41	4.76	4.35	4.07	3.86	3.71	3.59	3.49	3.41	3.34	3.29	3.24	3.20	3.16	3.13	3.10	2.60
4	224.58	19.25	9.12	6.39	5.19	4.53	4.12	3.84	3.63	3.48	3.36	3.26	3.18	3.11	3.06	3.01	2.96	2.93	2.90	2.87	2.37
5	230.16	19.30	9.01	6.26	5.05	4.39	3.97	3.69	3.48	3.33	3.20	3.11	3.03	2.96	2.90	2.85	2.81	2.77	2.74	2.71	2.21
6	233.99	19.33	8.94	6.16	4.95	4.28	3.87	3.58	3.39	3.22	3.09	3.00	2.92	2.85	2.79	2.74	2.70	2.66	2.63	2.60	2.10
7	236.77	19.35	8.89	6.09	4.88	4.21	3.79	3.50	3.29	3.14	3.01	2.91	2.83	2.76	2.71	2.66	2.61	2.58	2.54	2.51	2.01
8	238.88	19.37	8.85	6.04	4.82	4.15	3.73	3.44	3.23	3.07	2.95	2.85	2.77	2.70	2.64	2.59	2.55	2.51	2.48	2.45	1.94
9	240.54	19.38	8.81	6.00	4.77	4.10	3.68	3.39	3.18	3.02	2.90	2.80	2.71	2.65	2.59	2.54	2.49	2.46	2.42	2.39	1.88
10	241.88	19.40	8.79	5.96	4.74	4.06	3.64	3.35	3.14	2.98	2.85	2.75	2.67	2.60	2.54	2.49	2.45	2.41	2.38	2.35	1.83
11	242.98	19.40	8.76	5.94	4.70	4.03	3.60	3.31	3.10	2.94	2.82	2.72	2.63	2.57	2.51	2.46	2.41	2.37	2.34	2.31	1.79
12	243.91	19.41	8.74	5.91	4.68	4.00	3.57	3.28	3.07	2.91	2.79	2.69	2.60	2.53	2.48	2.42	2.38	2.34	2.31	2.28	1.75
13	244.69	19.42	8.73	5.89	4.66	3.98	3.55	3.26	3.05	2.89	2.76	2.66	2.58	2.51	2.45	2.40	2.35	2.31	2.28	2.25	1.72
14	245.36	19.42	8.71	5.87	4.64	3.96	3.53	3.24	3.03	2.86	2.74	2.64	2.55	2.48	2.42	2.37	2.33	2.29	2.26	2.22	1.69
15	245.95	19.43	8.70	5.86	4.62	3.94	3.51	3.22	3.01	2.85	2.72	2.62	2.53	2.46	2.40	2.35	2.31	2.27	2.23	2.20	1.67
16	246.46	19.43	8.69	5.84	4.60	3.92	3.49	3.20	2.99	2.83	2.70	2.60	2.51	2.44	2.38	2.33	2.29	2.25	2.21	2.18	1.64
17	246.92	19.44	8.68	5.83	4.59	3.91	3.48	3.19	2.97	2.81	2.69	2.58	2.50	2.43	2.37	2.32	2.27	2.23	2.20	2.17	1.62
18	247.32	19.44	8.67	5.82	4.58	3.90	3.47	3.17	2.96	2.80	2.67	2.57	2.48	2.41	2.35	2.30	2.26	2.22	2.18	2.15	1.60
19	247.69	19.44	8.67	5.81	4.57	3.88	3.46	3.16	2.95	2.79	2.66	2.56	2.47	2.40	2.34	2.29	2.24	2.20	2.17	2.14	1.59
20	248.01	19.45	8.66	5.80	4.56	3.87	3.44	3.15	2.94	2.77	2.65	2.54	2.46	2.39	2.33	2.28	2.23	2.19	2.16	2.12	1.57
21	248.31	19.45	8.65	5.79	4.55	3.86	3.43	3.14	2.93	2.76	2.64	2.53	2.45	2.38	2.32	2.26	2.22	2.18	2.14	2.11	1.56
22	248.58	19.45	8.63	5.79	4.54	3.86	3.43	3.13	2.92	2.75	2.63	2.52	2.44	2.37	2.31	2.25	2.21	2.17	2.13	2.10	1.54
23	248.83	19.45	8.64	5.78	4.53	3.85	3.42	3.12	2.91	2.75	2.62	2.51	2.43	2.36	2.30	2.24	2.20	2.16	2.12	2.09	1.53
24	249.05	19.45	8.64	5.77	4.53	3.84	3.41	3.12	2.90	2.74	2.61	2.51	2.42	2.35	2.29	2.24	2.19	2.15	2.11	2.08	1.52
25	249.26	19.46	8.63	5.77	4.52	3.83	3.40	3.11	2.89	2.73	2.60	2.50	2.41	2.34	2.28	2.23	2.18	2.14	2.11	2.07	1.51
26	249.45	19.46	8.63	5.76	4.52	3.83	3.40	3.10	2.89	2.72	2.59	2.49	2.41	2.33	2.27	2.22	2.17	2.13	2.10	2.07	1.50
27	249.63	19.46	8.63	5.76	4.51	3.82	3.39	3.10	2.88	2.72	2.59	2.48	2.40	2.33	2.27	2.21	2.17	2.13	2.09	2.06	1.49
28	249.80	19.46	8.62	5.75	4.50	3.82	3.39	3.09	2.87	2.71	2.58	2.48	2.39	2.32	2.26	2.21	2.16	2.12	2.08	2.05	1.48
29	249.95	19.46	8.62	5.75	4.50	3.81	3.38	3.08	2.87	2.70	2.58	2.47	2.39	2.31	2.25	2.20	2.15	2.11	2.08	2.05	1.47
30	250.10	19.46	8.62	5.75	4.50	3.81	3.38	3.08	2.86	2.70	2.57	2.47	2.38	2.31	2.25	2.19	2.15	2.11	2.07	2.04	1.46
∞	254.3	19.50	8.53	5.63	4.36	3.67	3.23	2.93	2.71	2.54	2.40	2.30	2.21	2.13	2.07	2.01	1.96	1.92	1.88	1.84	1.00

（2）t-检验　同样可以用式（3.36）定义的 t-统计量来检验一个预测的和一个实际观测的值是否有显著差异。

$$t=\left|\frac{Y-\hat{Y}}{S_e}\right| \tag{3.36}$$

式中，Y 是一个因变量的被观测测量值；$\hat{Y}$ 是一个预测值；S_e 是式（3.28）中定义的估计量的标准误差。

我们来假设实际的机器人速度为 36.2，有 11.3V 电压。这是否是实际测量值（36.2）和预测值（38.8）在 5% 水平上有显著的差异？从式（3.36）中，我们得到一个 $t=\left|\frac{36.2-38.8}{3.13}\right|=0.8$ 的值，从表 3.3 中我们得到一个临界值 2.776。0.8 在这个临界值之下，在预计值和实测值之间没有显著的差异。

3.6.2 皮尔森线性相关系数 r

式（3.37）给出了度量两个连续型变量 x 和 y 的相关性的线性相关系数 r（皮尔森的 r）：

$$r = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2}\sqrt{\sum_i (y_i - \bar{y})^2}} \tag{3.37}$$

式中，r 是皮尔森（Pearson）线性相关系数；$\bar{x}$ 和 $\bar{y}$ 是 $x_i s$ 和 $y_i s$ 的均值。

r 描述一个变量，此变量在 -1 与 $+1$ 之间（-1 表示完全负相关——即点都在一条斜率为负的完全直线上；$+1$ 表示完全正相关——即点都在一条斜率为正的完全直线上）。若 r 很小，说明 x、y 之间不存在很强的线性相关性。

1. 皮尔森 r 的显著性检验

任意两个数可能存在非零的 r，而我们要回答的问题是，一个确定的 r 是否表示统计上有显著的相关性。

表3.19给出在 $df = N - 2$（N 为数对的个数）个自由度下，皮尔森（Pearson）相关系数的显著性水平。如果我们对 r 是否显著大于或小于零感兴趣的话，我们必须使用双尾检验，否则只需使用单尾检验。

表3.19 皮尔森 r 的显著性水平

	单尾检验的显著性水平			
	0.05	0.025	0.01	0.005
df	双尾检验的显著性水平			
	0.10	0.05	0.02	0.01
1	0.988	0.997	0.9995	0.9999
2	0.900	0.950	0.980	0.990
3	0.805	0.878	0.934	0.959
4	0.729	0.811	0.882	0.917
5	0.669	0.754	0.833	0.874
6	0.622	0.707	0.789	0.834
7	0.582	0.666	0.750	0.798
8	0.549	0.632	0.716	0.765
9	0.521	0.602	0.685	0.735
10	0.497	0.576	0.658	0.708
15	0.412	0.482	0.558	0.606
20	0.360	0.423	0.492	0.537
25	0.323	0.381	0.445	0.487
30	0.296	0.349	0.409	0.449
40	0.257	0.304	0.358	0.393
60	0.211	0.250	0.295	0.325
80	0.183	0.217	0.256	0.284
100	0.164	0.195	0.230	0.254

2. 线性相关系数：Scilab 编程

下面的 Scilab 程序可以算出皮尔森（Pearson）线性相关系数。

```
function  [r] = pearson(a,b)
//  Ulrich  Nehmzow
//  计算两个时间序列 a 和 b 之间的线性相关系数(皮尔森 Pearson 的 r)

a = a-mean(a)
b = b-mean(b)
numerator = 0
for  i = 1: length(a)

        numerator = numerator + a(i) * b(i)
end
r = numerator/(sqrt(sum(a^2)) * sqrt(sum(b^2)))

//现在分析统计显著性
//注意:这个程序近似自由度超过 df = 60 的临界值时的函数
//如果程序结果为"close",结果需要与列表中的临界值作比较
rcrit = [0.997000 0.950000 0.878000 0.811000 0.754000 0.707000
0.666000 0.632000 0.602000 0.576000 0.553000 0.532000 0.514000
0.497000 0.482000 0.468000 0.456000 0.444000 0.433000 0.423000
0.413000 0.404000 0.396000 0.388000 0.381000 0.374000 0.367000
0.361000 0.355000 0.349000 0.344000 0.339000 0.335000 0.330000
0.325000 0.321000 0.317000 0.312000 0.308000 0.304000 0.301000
0.298000 0.294000 0.291000 0.288000 0.285000 0.282000 0.279000
0.276000 0.273000 0.271000 0.268000 0.266000 0.264000 0.262000
0.259000 0.257000 0.255000 0.252000 0.250000]

df = length(a) - 2
if  df < = 60
        critical = rcrit(df)
else
        critical = 0.195 + exp( - 0.05 * df)
end
printf("The  critical  value  for  statistical
            significance  is  %5.2f\n",critical)
if  abs(r) > critical
        printf("Therefore  the  computed  r  of
            %5.2f  is  significant(p < 5\%,two—tailed)\n",r)
else
```

```
        printf("Therefore the computed r of
            %5.2f is not significant(p>5\%,two—tailed)\n",r)
end
```

3. 线性相关系数（皮尔森的 r）：举例

3.6.1节中提到的例子中，我们想要知道机器人速度与电池电压之间是否相关，两者的皮尔森相关系数为 $r=0.823$。我们想要知道这个 r 是否在5%的显著性水平上是显著的。

由于我们想要知道 r 是显著大于零还是显著小于零，因此我们使用双尾检定。在 $df=N-2=4$ 时，我们得到临界值为0.811 。在这种情况下 r 刚好大于临界值，因此显著相关。

3.7 趋势的非参数检验

3.7.1 斯皮尔曼等级相关

3.6.2节介绍了相关分析，并把它作为一种衡量两个变量之间相关程度的方法，它基于数值，并假定其总体为正态分布。

但是，一些实验不能产生数值数据以供分析，相关分析就不能在这些情况下使用。但是所提供的与大小和评价相关的等级（秩）是可用的，相关分析就仍然可用。这就是本节要介绍的斯皮尔曼（Spearman）等级相关分析。

在只有秩是可用的情况下，这种秩分析是适用的。此外，它还有更大的优势。在计算方式上，它比基于数值的相关分析更为简洁。与基于数值的分析方法相比，数据中的极值对其的影响更小。在某些情况下，它同样适用于发现非线性相关。然而，如果可以用式（3.37）算出线性相关系数，那么就不要再用等级相关分析了。假设数据集之间没有相关性（$r_s=0$），在此假设下，斯皮尔曼等级相关检验算出一个等级相关系数 r_s。r_s 在 -1 与 $+1$ 之间，（-1 表示完全相关，为负斜率；$+1$ 表示完全相关，为正斜率）。如果存在显著相关，这意味着等级相关系数 r_s 与0有显著差异。两个数据集中的一个可以用来对另一数据集进行有意义的预测。

当数据不相关时，r_s 可以用式（3.38）算出[⊖]：

$$r_s = 1 - \frac{6\sum d^2}{n(n^2-1)} \qquad (3.38)$$

式中，d 表示每对观测值秩的差值；n 表示一共有多少对观测值。

⊖ 有结合秩会给方程3.38引入误差。

无论两组数据集是否是正态分布，式（3.38）都能使用。

1. 斯皮尔曼等级相关举例：保洁机器人

一个保洁机器人有四种不同的行为：沿弧线行走、沿直线行走、绕开障碍和墙壁行走。表3.20给出的数据显示在一个特别的实验中，机器人沿墙壁行走的频数，以及右撞传感器被触发的频数。图3.10为这些数据的散点图。问题是：这个关系是否与 $r_s=0$（无效假设）有显著差异。

为计算 r_d，我们对表3.20的两个纵列分别求秩，算出每对数的秩的差。这些表示在表3.21中。如所有的秩分析一样，结合秩可以看做平均秩（例如，第3和第4秩是结合秩，得到秩3.5，下一个秩为5）。

如果 $n=9$，我们会得到 $r_s=1-\frac{6\times 12.5}{9\ (9^2-1)}=0.896$。这个结果是否显然背离我们的无效假设 $r_s=0$？

表3.20 保洁机器人实验：机器人右撞缓冲器的信号与沿墙壁行走的关系

实验序号	右撞传感器的频率	沿墙壁行走的频率
1	18	4
2	12	3
3	21	6
4	13	3
5	22	7
6	40	9
7	38	12
8	8	4
9	41	12

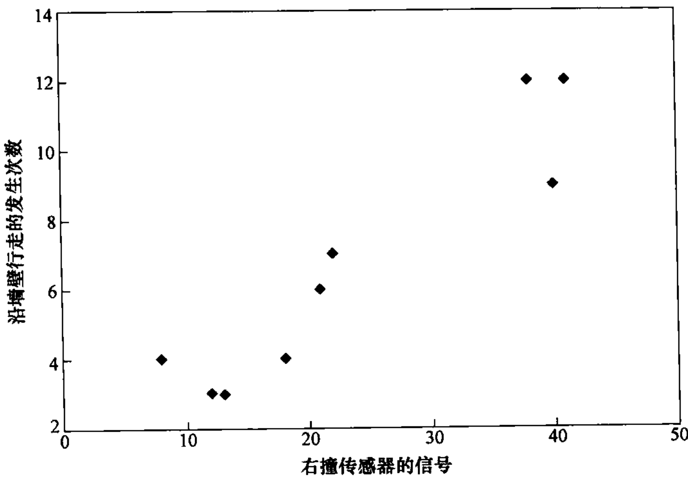


图3.10 表3.20数据的散点图

表 3.21 地面清洁机器人的实验：机器人右侧缓冲器和沿墙行为的信号之间的关系。(原始数据、秩与秩的差值)

实验次数	正确的缓冲次数	墙壁沿行次数	秩的次数	秩的 WF	d^2
1	18	4	4	3.5	0.25
2	12	3	2	1.5	0.25
3	21	6	5	5	0
4	13	3	3	1.5	2.25
5	22	7	6	6	0
6	40	9	8	7	1
7	38	12	7	8.5	2.25
8	8	4	1	3.5	6.25
9	41	12	9	8.5	0.25

总计：12.5

2. 斯皮尔曼等级相关：显著性检验

为了确定斯皮尔曼等级相关检验的显著性，我们可以应用以下两个标准之一：当 $n<30$，可以用表 3.22 来计算接受域。当 n 更大时，r_s 的抽样分布大致为正态分布，因此我们可以用表 3.2。

首先，我们看一下 $n<30$ 的情况，此时保洁机器人的情形，我们将分析上面给出的数据。

我们的无效假设是：在等级数据中没有统计相关性，另一个相反的假设是等级数据有相关性。我们将在 5% 的显著性水平上分析这些假设。因为我们要考虑因 r_s 过大或过小而超过我们期望的情况，那么双尾检验是最合适的方法。

表格 3.22 说明对于 $n=9$ 时的数据对，r_s 的临界值是 ±0.68。也就是，接受域的下限是 -0.68，上限是 0.68。我们计算的等级相关系数 $r_s=0.896$，这个值在接受域的外面，因此我们拒绝 $r_s=0$ 的无效假设，就是说在保洁机器人因碰撞而触发右侧缓冲器的传感器与机器人沿墙壁行走这两种行为之间的确存在统计相关性。

表 3.22 在双尾检验中斯皮尔曼等级相关系数 r_s 的临界值

(5% 的显著水平)(根据 [Levin and Rubin, 1980])。n 是数据对的标号

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
					0.90	0.83	0.75	0.71	0.68	0.64	0.61	0.58	0.55	0.53	0.52
n	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
	0.50	0.49	0.47	0.46	0.45	0.44	0.42	0.42	0.41	0.40	0.39	0.38	0.37	0.37	0.36

如果 $n>30$，则 r_s 的抽样分布数值基本上是正态的，我们可以使用标准差 σ_{r_s} 的公式（式 3.39），来计算 r_s 的接受区间。

$$\sigma_{r_s} = \frac{1}{\sqrt{n-1}} \tag{3.39}$$

举例来说，假设我们继续用这个保洁机器人再做31次实验，则 $n=10$。r_s 将等于0.53。

对于一个双尾检验，我们关心的是根据表3.2中 $z=1.96$ 时，等级相关系数在5%的显著水平上是否明显的高于或低于无效假设中的 $r_s=0$。我们的接受域的边界是 $0 \pm z\sigma_{r_s}$。对于这次试验，接受域的下限是 $-1.96 \times \frac{1}{\sqrt{40-1}} = -0.31$，接受域的上限是0.31。我们计算得到的 r_s 等于0.53，在接受域之外。这又一次证明 $r_s=0$ 的无效假设被拒绝。

3. 斯皮尔曼等级相关：Scilab 程序

下面的Scilab程序可以计算斯皮尔曼等级相关系数及作相关性的检验。

```
function[rs] = spearmant(x,y)
getf('ranking.sci')
//Ulrich  Nehmzow
//  计算非参数斯皮尔曼等级相关
//  向量 x 和 y 之间的

rankx = ranking(x)
ranky = ranking(y)
d = rankx-ranky
n = length(x)
rs = 1 - ((6 * sum(d^2))/(n^3 - n))
printf("Spearman  rank  correlation  coefficient:%f/n",rs)

//  现在对显著性进行分析
if  n<4
        printf("Too  little  data  for  significance  analysis\n")
        break
elseif  n<31
        critical = [999  999  999  999.90.83.75.71.68.64.61.58.55
.53.52.50.49.47.46.45.44.42.42.41.40.39.38.37.37.36]
        if(abs(rs) > critical(n))
                printf("This  coefficient  is  significant  (p<5\%)\n")
        else
                printf("This  coefficient  is  not  significant  (p>5\%)\n")
        end
else
        sigma = 1/sqrt(n - 1)
```

```
        if  abs(rs) >1.96 * sigma
                printf("This  coefficient  is  significant  (p<5\%)\n")
        else
                printf("This  coefficient  is  not  significant  (p>5\%)\n")
        end
end
function  [rnk] = ranking(a)
//Ulrich  Nehmzow
//  对向量 a 求秩
//  作为非参数统计检验的预处理

[s  k] = sort(a)

//  现在求等级向量
for  pos = 1:length(a)

//      r(k(pos)) = pos
        r(pos) = pos
end

//  现在检查结合秩,并相应调整 r-向量
pos = 1
while  pos < = length(a) - 1
        ties = 1
        while  s(pos) = = s(pos + ties)
                ties = ties + 1
                if  pos + ties > = length(a)
                        break
                end
        end
r(pos:pos + ties - 1) = mean(r(pos:pos + ties - 1))
pos = pos + ties
end

//  现在对原输入向量的位置重新评定等级
for  pos = 1:length(a)
        rnk(k(pos)) = r(pos)
end
```

3.8 分类数据分析

均值、标准差、t-检验和许多其他的统计分析方法只能用于分析连续的数据。但是，在机器人实验中，很多实验得到的结果是分类数据，例如，在分类系统中，任务是将得到的数据分为一类或几类。在本节中，我们将介绍分析这些分类数据的方法。

名义变量的定义是组成元素不能排序的变量，比如："颜色"、"味道"，在这里我们认为实验结果是名义变量。

接下来，我们感兴趣的是两个名义变量之间是否有联系。这种想法是很合理的，比如在分类任务中，哪个是输入变量，哪个是输出变量。在这种情况下，我们的问题是"分类器的输出是否与输入的信号相关"，换句话说就是"分类器是否合适?"。

两个变量的数据可以记录在一个列联表中，在表中我们可以做交叉分析。例如，如果有一场机器人比赛，在比赛中，三个机器人在三种不同的项目上进行多次比赛。在列联表中将会显示每一个机器人赢得每一场比赛的频率，同时，交叉分析将会给出机器人赢得比赛与项目之间是否存在关系。这将证实是否有机器人特别善于某一特定的项目。表 3.23 显示了这一分析过程的列联表。

表 3.23 一个列联表的例子（$n_{A,X}$是机器人 X 赢得比赛 A 的次数，$N_{.A}$是比赛 A 的总次数，$N_{Z.}$ 是机器人 Z 赢得比赛的总次数，以此类推）

	比赛 A	比赛 B	比赛 C	
机器人 X	$n_{A,X}$	$n_{B,X}$	…	
机器人 Y			…	
机器人 Z	…	…	…	$N_{Z.}$
	$N_{A.}$	$N_{.B}$	$N_{.C}$	N

3.8.1 确定两个变量之间的关系（χ^2检验）

确定两个变量之间的相关性的检验是χ^2 检验。

用 N_{ij}表示 x 的值等于 i，y 的值等于 j 的事件的数量。用 N 表示事件的总数量。用 $N_{i.}$ 表示不考虑 y 的值，x 的值等于 i 的事件的数量；用 N_{*j}表示不考虑 x 的值，y 的值等于 j 的事件的数量。

$$N_{j.} = \sum_j N_{ij}$$

$$N_{.j} = \sum_i N_{ij}$$

$$N = \sum_i N_{i.} = \sum_j N_{.j}$$

1. 得到期望值表

在χ^2检验中的无效假设表示两个变量 x 和 y 之间没有显著相关性。为了检验这个无效假设，需要确定期望值，得到无效假设正确时的期望值。这个期望值可以用一般的应用软件或者通过下面的推导来得到。

在一个表格中，比如表3.23，$\frac{n_{ij}}{N_{.j}}$是事件 i 发生的概率估计，对于给定的 j，就有$\frac{n_{ij}}{N_{.j}} = p\ (i|j)$。如果无效假设是正确的，对于一个确定的数值 i，给出一个特定的值 j 时的概率应该与忽略 j 时，$x = i$ 所对应的概率是相同的，就是说$\frac{n_{ij}}{N_{.j}} = p\ (i|j) = p\ (i)$。

已知$p\ (i) = \frac{N_{i.}}{N}$也是正确的。在无效假设正确的假定下，我们可以推断

$$\frac{n_{ij}}{N_{.j}} = \frac{N_{i.}}{N} \tag{3.40}$$

其产生的期望表 n_{ij} 为：

$$n_{ij} = \frac{N_{i.} N_{.j}}{N} \tag{3.41}$$

χ^2 分布的定义为式（3.42）

$$\chi^2 = \sum_{i,j} \frac{(N_{ij} - n_{ij})^2}{n_{ij}} \tag{3.42}$$

计算χ^2分布（式3.42）连同$\chi^2_{.05}$的概率函数（表3.24）可以用来确定在变量 i 与变量 j 之间是否有相关性。在一个 I 和 J 的表格中，自由度 m 等于

$$m = IJ - I - J + 1 \tag{3.43}$$

如果$\chi^2 > \chi^2_{.05}$（见表3.24），则 i 和 j 之间具有相关性

表3.24 相关度为0.1，0.05，0.1时的临界的χ^2分布数值

自由度	10%	5%	1%
1	2.71	3.84	6.63
2	4.61	5.99	9.21
3	6.25	7.81	11.34
4	7.78	9.49	13.28
5	9.24	11.07	15.09
6	10.64	12.59	16.81

（续）

自由度	10%	5%	1%
7	12.02	14.07	18.48
8	13.36	15.51	20.09
9	14.68	16.92	21.67
10	15.99	18.31	23.21
11	17.28	19.68	24.72
12	18.55	21.03	26.22
13	19.81	22.36	27.69
14	21.06	23.68	29.14
15	22.31	25.00	30.58
16	23.54	26.30	32.00
17	24.77	27.59	33.41
18	25.99	28.87	34.81
19	27.20	30.14	36.19
20	28.41	31.41	37.57

如果 m 大于 30，相关性可以用 $\sqrt{2\chi^2}-\sqrt{2m-1}$来计算。如果计算的结果超过 1.65，则 i 与 j 之间是相关的。

如果不使用表 3.24，χ^2 分布的值同样可以用软件计算。在 scilab 程序中的实现方式是：

cdfchi（"X"，DF，Q，P）

DF 为自由度的数值，P 是所需的相关性的大小，Q = 1 − P

2.χ^2 统计的实际考虑

为了使χ^2 统计的数值可信，需要对数据做很好的处理。在下面的情况下，有两个经验法则。

1. 在期望值表格 n_{ij}中，所有的单元格的数值应该不小于 1。在 $m\geqslant 8$ 且 $n\geqslant 40$ 时，单元格的数值应该不小于 4（［Sachs，1982，p. 321］）。

2. 在期望值表格 n_{ij}中，值不超过 5 的单元格的比例应该不多于 5%

如果不满足了上述的任意一个条件，可以通过合并列联表的行或列，以满足上述的准则。

3.χ^2 检验的例子：自定位评价

在一个具有四个明显的地标 A、B、C、D 的环境中放置一个移动机器人。机器人的地标辨识程序产生四个反馈 α、β、γ 和 δ，将反馈得到的这四个位置给接收器。在这个实验中共有 200 次随机地碰到这四个不同的地标。

数据包含在表3.25中（数值表示一个特定的地标对应于一个特定的反馈编号的次数）。

表3.25 地标—辨识程序的列联表

	α	β	γ	δ	
A	19	10	8	3	$N_{A.}=40$
B	7	40	9	4	$N_{B.}=60$
C	8	20	23	19	$N_{C.}=70$
D	0	8	12	10	$N_{D.}=30$
	$N_{.\alpha}=34$	$N_{.\beta}=78$	$N_{.\gamma}=52$	$N_{.\delta}=36$	$N=200$

分类器的输出是否与机器人的位置显著相关？

解答：根据式（3.41）$n_{A\alpha}=\dfrac{40\times 34}{200}=6.8$，$n_{A\beta}=\dfrac{40\times 78}{200}=15.6$

其他的计算与此类似（期望值表格为表3.26）

期望值表格为χ^2分析做了很好的处理；没有数值小于4的单元格。

根据式（3.42），$\chi^2=\dfrac{(19-6.8)^2}{6.8}+\dfrac{(10-15.6)^2}{15.6}+\cdots=66.9$。这个系统有16－4－4＋1＝9个自由度（根据式3.43）。根据式3.24，则有：$\chi^2_{0.05}=16.9$

$$\chi^2=66.9>\chi^2_{0.05}=16.9 \tag{3.44}$$

因此，机器人位置和定位系统的输出显著的相关。

表3.26 期望值表

	α	β	γ	δ
A	6.8	15.6	10.4	7.2
B	10.2	23.4	15.6	10.8
C	11.9	27.3	18.2	12.6
D	5.1	11.7	7.8	5.4

3.8.2 确定相关性的强弱：克莱默 V 参数

在统计学里，χ^2检验是一种应用非常普遍的检验方法，也是由于这个原因，其表达能力有限。实际上，如果随机事件表里的样本数量足够大的话，该检验通常会表明变量之间的一种重要的相互关系。这和检验能力有关，如果有足够数量的样本是可用的，这种能力还能将有更高显著性水平上的较小的相关性放大。鉴于此，最好能再将χ^2进行参数化，以使其独立于样本的大小。这将使我们能够确定一种关系的强弱程度，并且能够和另一个随机变量表进行比较。

克莱默V参数重新定义了χ^2，将其限制在区间$0\leqslant V\leqslant 1$，$V=0$表示x和y之间没有联系，$V=1$表示完全相关。克莱默V参数由式（3.45）给出：

$$V = \sqrt{\frac{\chi^2}{N\min(I-1, J-1)}} \tag{3.45}$$

式中，N 为大小为 $I \times J$ 的随即变量表中样本的总数；min（$I-1$，$J-1$）是 $I-1$ 与 $J-1$ 中的较小值。

假设已给出随机变量表 m，下面的 Scilab 代码将计算出 x^2 与克莱默 V：

```
function  [x2] = chisq(m)
//  Ulrich  Nehmzow
//对列联表 m 进行变卡方(Chi squared)分析

[r  c] = size(m)
N = sum(m)

//计算表格的期望值
    for  i = 1:r
            for  j = 1:c
                    n(i,j) = sum(m(:,j)) * sum(m(i,:))/N
            end
end
//现在计算对卡方(Chi squared)的作用
for  i = 1:r
            for  j = 1:c
                    xcontrib(i,j) = ((m(i,j) - n(i,j))^2)/n(i,j)
            end
end
X2 = sum(xcontrib)
[p  sig] = cdfchi("PQ",X2,r*c—r—c+1)
if  (sig>0.05)
            printf("There  is  no  significant  correlation
            between  the  two  variables(p>0.05)\n")
else
            printf("There  is  a  significant  correlation
            between  the  two  variables  (p<%4.3f)\n",sig)
end
v = sqrt(X2/(sum(m)*(-1+min(size(m)))))
printf("Cramer"s  V = %f\n",v)
```

例：克莱默（Crammer）V 参数

下面来比较两个地图创建的例子。例 A 给出的随机变量表如表 3.27 所示，

例 B 给出的表如表 3.28 所示。

表 3.27　地图创建 1 的结果

	α	β	γ	δ	
A	29	13	5	7	$N_{A.}=54$
B	18	4	27	3	$N_{B.}=52$
C	8	32	6	10	$N_{C.}=56$
D	2	7	18	25	$N_{D.}=52$
	$N_{.\alpha}=57$	$N_{.\beta}=56$	$N_{.\gamma}=56$	$N_{.\delta}=45$	$N=214$

表 3.28　地图创建 2 的结果

	α	β	γ	δ	ε	
A	40	18	20	5	7	$N_{A.}=90$
B	11	20	35	10	3	$N_{B.}=79$
C	5	16	10	39	5	$N_{C.}=75$
D	2	42	16	18	9	$N_{D.}=87$
E	6	11	21	9	38	$N_{E.}=85$
	$N_{.\alpha}=64$	$N_{.\beta}=107$	$N_{.\gamma}=102$	$N_{.\delta}=81$	$N_{.\varepsilon}=62$	$N=416$

问题：两种创建方法中的哪一种所产生的地图在机器人位置与地图反馈之间的相关性更强？

我们用克莱默 V 参数来回答这个问题。

表 3.29 和表 3.30 分别给出了两个表的期望值。通过观察两个表中的期望值数据，可以发现数据符合前面列出的标准。

表 3.29　地图创建 1 的期望值

	α	β	γ	δ
A	14.4	14.1	14.1	13.4
B	13.9	13.6	13.6	10.9
C	14.9	14.7	14.7	11.8
D	13.9	13.6	13.6	10.9

由地图创建 1，我们得到 $x^2=111$，$V=0.42$；由地图创建 2，我们得到 $x^2=229$，$V=0.37$。地图 1 中地图反馈和（机器人）位置的相关性更强。然而，两种实验都易受一些随机变动的影响，因此有必要将每个实验重复几次，以排除随即噪声的影响。

表 3.30　地图创建 2 的期望值

	α	β	γ	δ	ε
A	13.8	23.1	22.1	17.5	13.4
B	12.2	20.3	19.4	15.4	11.8
C	11.5	19.3	18.4	14.6	11.2
D	13.4	22.4	21.3	16.9	13
E	13.1	21.9	20.8	16.6	12.7

3.8.3 用基于熵的测量方法确定相关性的强弱

借助卡方（Chi squared）检验与克莱默（Crammer）的 V，我们可以确定一个随机列联表的行与列之间是否存在显著的联系。

然而，我们还需要一些测量联系强度的方法。下面介绍了两种定量的测量一个联系强度的方法。

我们脑海中特定的情景是这样的：一个移动机器人探索它的环境，创建出地图，然后用地图进行定位。

每一时刻，机器人都处在某个物理位置 L，它的定位系统会产生一个特定的反馈，表明机器人占据了世界上的某个位置。在一个好的定位系统中，L 和 R 之间的联系很强，而在一个基于随机猜测的定位系统中，L 和 R 没有联系。

基于熵的测量方法，尤其是熵 H 和不确定度 U，可以用来测量联系的强度。它们的确定（过程）如下。

1. 用熵来确定名义变量间联系的强度

在表 3.31 所给的例子中，采集了一个含有 100 个数据点的样本。每个数据点有两个特征：一个相当于由机器人预测的位置（机器人的反馈 R），另一个相当于观察者测得的机器人的实际位置（机器人的真实位置 L）。比如说，表 3.31 给出了一个含有 19 个数据点的单元，其中，机器人反馈的测量值为第三行，位置为第五列。

表 3.31 例子列联表（行相当与由特定的定位系统产生的反馈，列相当于由观察着所测得的机器人的真实位置。这个表格代表了 100 个数据点，也给出了每行每列的总数）

位置 L

R	0	2	15	0	1	18
e	10	10	0	0	0	20
S	0	2	1	0	19	22
P	5	7	3	1	1	17
.	0	0	0	23	0	23
(R)	15	21	19	24	21	100

为了分析列联表，需要先根据式（3.46）、式（3.47）和式（3.48）计算出所有的行、列以及整个表的总数 $N_{r.}$，$N_{.l}$和 N，其中 $N_{.r}$是各个反馈 r 的总数，$N_{.l}$是各个位置 l 的总数，N 是包含在行 r 和列 l 里的数据点的总数：

$$N_{r.} = \sum_{l} N_{rl} \qquad (3.46)$$

$$N_{.l} = \sum_{r} N_{rl} \quad (3.47)$$

$$N = \sum_{r,l} N_{rl} \quad (3.48)$$

行概率 $p_{r.}$，列概率 $p_{.l}$和单元概率 p_{rl}则根据式（3.49）、式（3.50）、式（3.51）算出：

$$p_{r.} = \frac{N_{r.}}{N} \quad (3.49)$$

$$p_{.l} = \frac{N_{.l}}{N} \quad (3.50)$$

$$p_{rl} = \frac{N_{rl}}{N} \quad (3.51)$$

L 的熵 $H(L)$，R 的熵 $H(R)$ 以及 L，R 的共同的熵 $H(R, L)$ 分别由式（3.52）、式（3.53）和式（3.54）给出：

$$H(L) = -\sum_{l} p_{.l} \ln p_{.l} \quad (3.52)$$

$$H(R) = -\sum_{r} p_{.r} \ln p_{.r} \quad (3.53)$$

$$H(L,R) = -\sum_{r,l} p_{rl} \ln p_{rl} \quad (3.54)$$

当运用式（3.52）、式（3.53）和式（3.54）时，须牢记 $\lim_{p\to 0} p\ln p = 0$。

对于上述情形，我们想解答的最重要的问题是“给出一个特定的反馈 R 和机器人定位系统，我们对机器人的当前位置的确定程度如何?”。这就是给定 R 后 L 的熵，$H(L \mid R)$。另一方面，如果一个特定的位置在不同的访问的基础上得出了不同的反馈 $R1$ 和 $R2$，我们不关心。机器人自我定位的关键是每一个反馈 R 都与一个精确的位置 L 紧密地联系在一起。

$H(L \mid R)$ 求得如下：

$$H(L \mid R) = H(L,R) - H(R) \quad (3.55)$$

$$\text{其中} \quad 0 \leqslant H(L \mid R) \leqslant H(L) \quad (3.56)$$

最后一条性质（式 3.56）指出 $H(L \mid R)$ 值的范围依赖于环境的大小，因为 $H(L)$ 随着位置空间的增长而增长。

2. 借助不确定性系数确定名义变量之间联系的强弱

熵 H 是一个介于 0 和 $\ln N$ 之间的数，其中 N 是数据点的数目。N 为 0 表示 L 和 R 完全相关，即每一个反馈 R 都确切地表明世界上的一个位置 L，熵 H 越大，L 和 R 之间的联系越弱。

不确定性系数 U 提供了另一种表达一个随机变量表中行变量与列变量之间联系强弱的方法，而且它有两个很好的性质：1）U 总是介于 0 和 1 之间，与随机变量表的大小无关，这使得不同大小的表进行对比。2）不确定性系数为 0 表

示没有联系，为1表示完全相关。这在直观上是大致“正确”的方法（联系越大，数值越大）。

给出 R 后，L 的不确定性系数 $U(L|R)$ 由式（3.57）给出：

$$U(L|R) \equiv \frac{H(L) - H(L|R)}{H(L)} \tag{3.57}$$

$U(L|R)$ 为0表示 R 没有提供关于 L 的有效信息，暗示机器人的反馈不能预测它的真实位置；$U(L|R)$ 为1表示 R 提供了所有所需的关于 L 的信息，表明反馈总是可以预测真实位置。另外必须注明的是，随机变量表里行与列的次序与计算结果没有影响。

对于一般的对称的情形，不确定性系数 $U(x, y)$ 由式（3.58）给出：

$$U(x,y) = 2\frac{H(y) + H(x) - H(x,y)}{H(x) + H(y)} \tag{3.58}$$

3. 计算不确定性系数的例子

一个机器人定位系统产生了如表3.31所示的反馈。在系统反馈与机器人位置之间是否存在统计意义上的显著相关性？

为了回答这个问题，我们根据式（3.57）来计算不确定性系数 $U(L|R)$。为此，我们需要计算 $H(L)$、$H(R)$ 和 $H(L|R)$。

通过应用式（3.52）、式（3.53）、式（3.54）和式（3.57），我们得到

$$H(L) = -\left(\frac{15}{100}\ln\frac{15}{100} + \frac{21}{100}\ln\frac{21}{100} + \cdots + \frac{21}{100}\ln\frac{21}{100}\right) = 1.598$$

$$H(R) = -\left(\frac{18}{100}\ln\frac{18}{100} + \frac{20}{100}\ln\frac{20}{100} + \cdots + \frac{23}{100}\ln\frac{23}{100}\right) = 1.603$$

$$H(L, R) = -\left(0 + \frac{2}{100}\ln\frac{2}{100} + \frac{15}{100}\ln\frac{15}{100} + \cdots + \frac{23}{100}\ln\frac{23}{100} + 0\right) = 2.180$$

$$H(L|R) = 2.180 - 1.603 = 0.577$$

$$H(L|R) = \frac{1.598 - 0.577}{1.598} = 0.639$$

该不确定系数显示了定位系统的反馈与机器人实际位置之间有比较密切的相关性。

以下为 scilab 程序计算不确定系数的过程。

```
function [u] =uc (m)
// Ulrich Nehmzow
// 计算 [x, y] 的列联表 m 的不确定性系数 U (x, y)
[x y] =size (m)
```

```
n = sum (m)

// 计算必要的概率
for i = 1: x
        pi (i) = sum (m (i, :))
end
pi = pi/n

for j = 1: y
        pj (j) = sum (m (:, j))
end
pj = pj/n
pij = m/n

//现在计算熵
hx = 0
for i = 1: x
        if (pi (i) ~ =0)
            hx = hx - pi (i) * log (pi (i))
        end
end
hy = 0
for j = 1: y
        if (pj (j) ~ =0)
            hy = hy - pj (j) * log (pj (j))
        end
end
hxy = 0
for i = 1: x
        for j = 1: y
            if (pij (i, j) ~ =0)
                hxy = hxy - pij (i, j) * log (pij (i, j))
            end

        end
end
// 现在计算不确定性系数
u = 2 * ((hy + hx - hxy) / (hx + hy))
```

3.9 主成分分析法

到目前为止，本章节已经讨论了如何以统计方法对数据进行分析和比对。但在很多情况下，我们感兴趣的并不是两组数据集是否不同，而是针对单个数据集，对其数据结构及性质有更深入的了解。在这样的情况下，重新整合数据将隐含数据模式显现出来，如通过恰当的坐标变换，对分析过程将有很大的帮助。主成分分析法（PCA）就是这样的方法。

主成分分析法不仅可以确定数据模式，还可以用来进行数据压缩。该法实质就是坐标变换，即重新整合原始数据，使数据沿新定义的坐标中标准差最大。这些新坐标即为主成分。

数据分布如图 3.11 和表 3.32 所示。

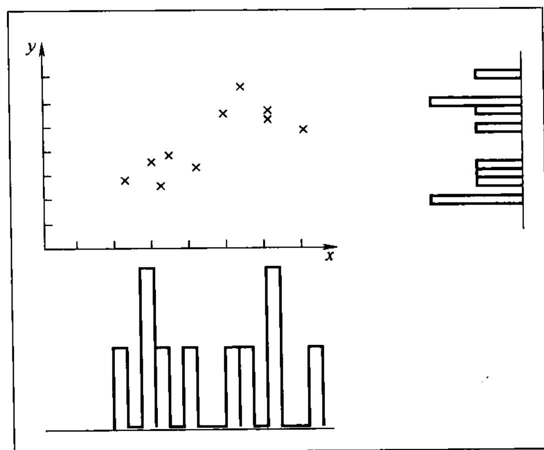


图 3.11　对题例数据进行主成分分析（柱状图显示了数据沿 x 轴和 y 轴的分布）

表 3.32　图 3.11 中数据数值

x	0.9	1.2	1.7	1.3	1.4	2.2	2.5	2.	2.9	2.5
y	1.1	1.4	1.3	1.	1.5	2.6	2.2	2.2	1.9	2.1

显然，用肉眼都可以看出数据分为两组，但如果图 3.11 所示使用坐标系（x，y），数据组 x 和数据组 y 都无法明确确定数据点应属哪一组。我们可以通过坐标变换轻易区分出两个组，如图 3.12 所示。

在进行主成分分析之前，数据集必有零均值，所以我们首先分别用 x 和 y 减去 x 的均值和 y 的均值。

主成分—我们将数据组降维变换后的坐标系，即为协方差矩阵的特征向量。所以我们需要确定协方差数据，并计算出特征向量和特征值。

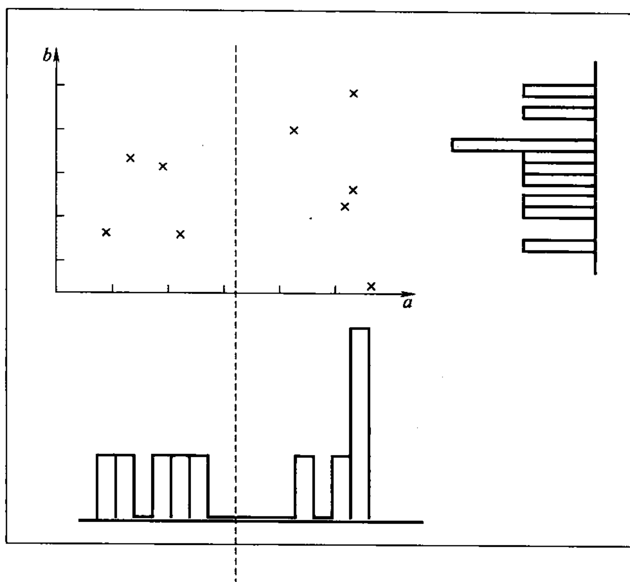


图 3.12 与图 3.11 为相同的一组数据，但是通过不同的坐标（主成分）显示出来（坐标轴 a 为第一主成分，坐标轴 b 为第二主成分。如果只使用坐标 a，两个组可以很容易区分开来）

协方差矩阵 $\boldsymbol{C}$ 的定义如式（3.59）所示，协方差值如式（3.60）所示。

$$\boldsymbol{C}=\begin{pmatrix}\operatorname{cov}(x,x) & \operatorname{cov}(x,y) & \operatorname{cov}(x,z) & \cdots \\ \operatorname{cov}(y,x) & \operatorname{cov}(y,y) & \operatorname{cov}(y,z) & \cdots \\ \operatorname{cov}(z,x) & \operatorname{cov}(z,y) & \operatorname{cov}(z,z) & \cdots \\ \cdots & \cdots & \cdots & \cdots\end{pmatrix} \tag{3.59}$$

$$\operatorname{cov}(x,y)=\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{n-1} \tag{3.60}$$

式中，$\bar{x}$，$\bar{y}$分别表示 x，y 的均值；n 表示 x，y 中数据点的数量；协方差 cov（x，y）说明 x，y 值是同时增长（正协方差）还是同时减小（负协方差），或者是该两维数据互相独立（协方差值为零）。

协方差矩阵包括所有可以从 n 维数据计算的$\frac{n!}{2\ (n-2)!}$个协方差矩阵 $\boldsymbol{A}$ 的特征值和特征向量的定义如式（3.61）所示：

$$(\boldsymbol{A}-\lambda_j\boldsymbol{I}_n)w_j=0 \tag{3.61}$$

式中，$\boldsymbol{I}_n$ 为单位矩阵，单位矩阵对角线上值为 1，非对角线各处计算出的值为零。

对于计算大于三行三列的矩阵的特征值和特征向量，用式（3.61）手工计算比较困难，我们最好用数学分析软件。

在 Scilab 分析软件中，可以用协方差矩阵（covarmat）表示如下

covarmat = [1 * corr (xa, xa, 1) / (1 - 1) 1 * corr (xa, ya, 1) / (1 - 1);
1 * corr (ya, xa, 1) / (1 - 1) 1 * corr (ya, ya, 1) / (1 - 1)]

其中，xa，ya 为均值为零的原始数据，1 为 xa，ya 中数据点的个数。

特征值可以由命令 Eigenvalues = spec (covarmat) 算出，而特征向量可以由 [eigenvalues, pcas, bs] = spec (covarmat) 确定。

以下程序可以算出二维数据的特征值和特征向量，并将原始数据进行坐标变换至 PCA 区域。

```
function[pcone,pctwo] = pcc(x,y)
//  (c)  Ulrich  Nehmzow
//  计算二维数据 x 和 y 的主成分
//  将原始数据从笛卡尔空间转换到含两个主成分的空间

xbasc()
subplot(211)
plot2d(x,y, -2)
xtitle('Original  Data','X','Y')

//  首先将数据转换为零均值数据
xa = x - mean(x)
ya = y - mean(y)

//  现在计算协方差矩阵
1 = length(xa)
covarmat = [1 * corr(xa,xa,1)/(1 - 1)1 * corr(xa,ya,1)/(1 - 1);
1 * corr(ya,xa,1)/(1 - 1)1 * corr(ya,ya,1)/(1 - 1)]

//  计算特征值
eigenvalues = spec(covarmat)

//  计算特征向量,即主成分
[eigenvalue,pcas,bs] = bdiag(covarmat)
printf("One  princtipal  component  is")
printf("%f%f(EV = %f)\n\n\n",pcas(1,1),pcas(2,1),eigenvalue(1,1))
printf("The  other  principal  component  is")
printf("\n%f%f(EV = %f)",pcas(1,2),pcas(2,2),eigenvalue(2,2))

p1 = pcas(:,1)
```

```
p2 = pcas(:,2)
if   eigenvalue(1,1) < eigenvalue(2,2)
          temp = p1
          p1 = p2
          p2 = temp
end

//  现在重新计算坐标值
for   i = 1: length(x)
          pcone(i) = [x(i);y(i)]' * p1
          pctwo(i) = [x(i);y(i)]' * p2
end

subplot(212)
plot2d(pcone,pctwo, -2)
xtitle(' Data   after   coordinate   transform ',' PC   1 ',' PC2 ')
```

将该程序应用到表3.32中的数据，可得到以下结果：

```
pcc (x, y);
One  principal  component  is
0.791798   0.610783    (EV =0.650954)

The  other  principal  component  is
-0.610783   0.791798    (EV =0.080713)
```

如上所述，主成分分析法可以用来确定数据模式（例如将第一主成分数据显现）以及数据压缩（有损耗）。在图3.11和图3.12所给数据中，数据压缩可以通过使用第一主成分来完成，即仅使用一个坐标系，而非原始的两个坐标系。

加深阅读：

- Rechard Levin and David Rubin, Applied elementary statiscitics, Prentice Hall 1980
- Allen Edwards, Statistics methods, Holt, Rinehart and Winston 1967
- Chris Barnard, Francis Gilbert and Peter McGregor, Asking questions in biology, Longman 1993

第4章　动力系统理论和智能体行为

概要：本章介绍动力系统理论的概念，例如，相空间的重构以及它们在智能体行为中的应用。

4.1　引言

如果一个（物理的）系统的行为描述是依赖于时间的，则这个系统被称为动力系统。它的行为可以用统计学方法及数学公式来描述（特别是用差分或微分方程）：系统的行为可以数学表达为“状态空间”或相空间中的运动，也就是系统描述子的多维空间中运动。移动机器人的行为很显然是与时间相关的，也就是说移动机器人是一个动力系统。因此，应用动力系统理论去分析机器人—环境交互作用是很有趣的。这就是本章的主要内容。

4.2　动力系统理论

动力系统原理是利用数学的方法描述动力系统，尝试用微分方程描述复杂动力系统的运动（例如智能体与其环境的交互）。广义来讲，这种行为是智能体在空间中的运动，这里的“空间”指的是物理空间，但是它也可以指其他的空间，例如4.2.1节中将介绍的理论上的相空间。动力系统理论的概念甚至已经扩展到了透过“认知空间”去描述智能体的运动，并试图数学地描述认知推理。

作为一门学科，动力系统理论包含许多方法使智能体在空间中的运动更形象化，并可进行数学的分析、分类、测量和描述。因此，对于第2章简要提到的数学地分析并描述智能体（如移动机器人）行为，动力系统理论将是实现这一目标的理想工具。

4.2.1　相空间

“相空间”是一个用来描述动力系统的所有可能状态空间的术语。

在相空间中的一个特定的状态可以充分地描述一个系统，并且它包含了对动力系统的下一个状态作出预测所需的所有信息。

1. 理想单摆的相空间

例如理想单摆，有一个自由度——沿摆动方向的圆弧角 ϕ，单摆的位置为

$\phi(t)$，单摆速度为 $\dot{\phi}(t)$，它完整地描述了各个时刻单摆的运动状态。因此，理想单摆的相空间是一个由 $\phi(t)$ 和 $\dot{\phi}(t)$ 定义的二维空间，单摆的物理运动可以通过相空间的运动完整地描述。

理想单摆的相空间是椭圆形的（见图 4.1）。随着单摆在物理空间中向负方向和正方向摆动，它在相空间中的 $(\phi,\dot{\phi})$ 坐标系也随之移动：从 $(\phi_{max},0)$ 开始，经过 $(0,-\dot{\phi}_{max})$、$(-\phi_{max},0)$ 和 $(0,\dot{\phi}_{max})$ 最后回到 $(\phi_{max},0)$。

通过相空间的轨迹（轨道）在单摆的情况下是椭圆（见图 4.1），这个轨迹被命名为"吸引子"，因为动力系统将遵循那个特殊轨迹，而不管初始条件，它都会通过相空间被"吸引"到那个轨迹上。

2. 哈密尔顿系统的相空间

与哈密尔顿（Hamiltonian）力学相关的动力系统的相空间，是用沿着该系统每个自由度的位置 $z(t)$ 和动量 $p(t)=mv(t)$（v 是速度）来定义的。这是对"相空间"和"自由度"的严格定义；然而"自由度"有时可以简单地理解为"相空间的一个坐标系"。除非另有说明，本书中的"自由度"均指"相空间的一个坐标系"。

如果一个哈密尔顿（Hamiltonian）系统有 n 个自由度，它的相空间将有 $2n$ 个维度、位置和动量。

3. 移动机器人的自由度

一个系统有多少个自由度并不总是那么显而易见，因而它的相空间有多大也不好简单表达。幸运的是，重构吸引子并不需要这些知识（参见 4.2.3 节内容）。

在这个移动机器人进入一个二维平面的例子中，可以说自由度对于机器人还是有用的。接着，我们将会检验移动机器人的四种基本类型：一个全向移动机器人（我们称之为"球"，因为它的运动和球有很多共同之处）；一个差动式机器人；一个可以做阿克曼转向的机器人（我们将会称它为"车"，因为它的运动和传统汽车很相似）；以及一个履带式机器人（称为"火车"）。这四种类型的机器人如图 4.2 所示。

前三种机器人可以在空间中取任何位置和方向（x，y，φ），但取得它们的方法都是不一样的。

我们定义一个全自由度作为一个轴（x、y 或 φ），它可以取任何的位置，且不需要依赖于其余两个轴。等价的定义就是一个全自由度是这样一个轴，在这个轴上任何外力作用都可以不受阻力（认为它是理想机器人）。

根据这些定义，在"球"的例子中，我们可以看到外力可以在三个轴上单

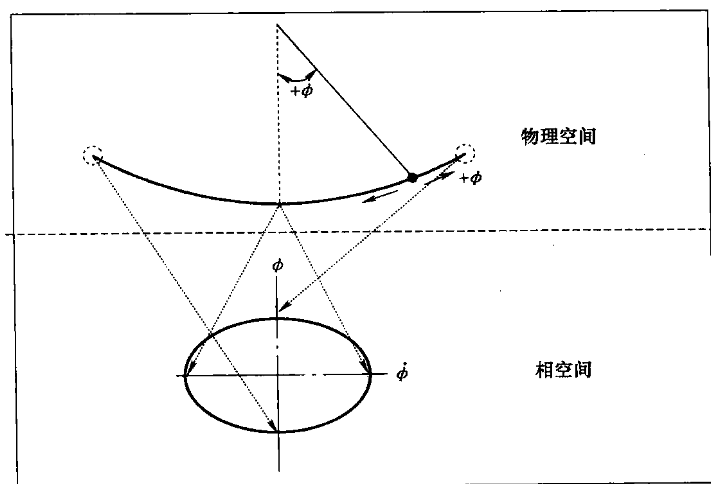


图 4.1 理想钟摆的物理空间和相空间
（箭头指出了物理空间和相空间的相互联系）

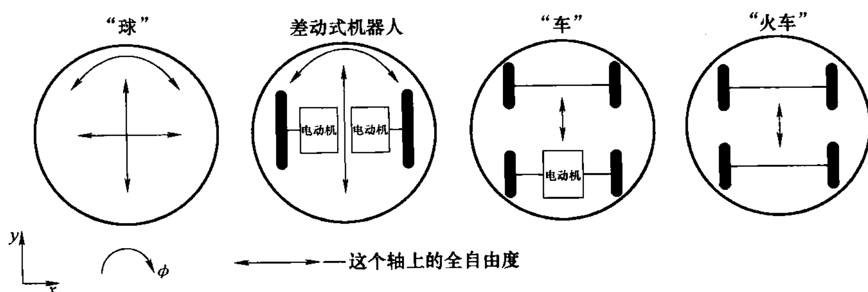


图 4.2 不同类型的机器人驱动系统及其全自由度

独地作用，并且可以在不改变其他两个轴的情况下单独改变一个轴的坐标。因此，“球”有三个自由度。

差动式机器人能沿着 y 轴和 φ 轴自由地改变它的坐标，但是如果机器人想要改变 x 轴的坐标，它就必须同时改变 φ 轴坐标。因此，它有两个全自由度（y 和 φ），但是因为机器人也能够改变 x 轴的坐标，所以它拥有超过两个（少于三个）的自由度 。通常这被称为“2.5 个自由度”。

在“车”的例子里，它能沿着 y 轴自由地改变它的坐标（一个自由度）。但是如果想要沿 φ 轴改变坐标，y 轴的坐标也要改变，因此这不是全自由度，而要改变 x 轴坐标，φ 轴也要随之改变（因此也需要改变 y 轴的位置），所以这也不是一个全自由度。那“车”有几个自由度呢？也许可以说是“一个加两个半个”？

最后来讲“火车”，它只有一个全自由度——y，而 x，φ 两个轴不能自由取值，因为它们已经预先被轨道所确定。它因此只有一个自由度，即它的相空间是二维的（y 和 $\dot{y}$）。

4.2.2　例证：通过重构相空间，分析机器人的运动

下面所给的假设的例子是为了举例说明这本书后面将提及的方法，也为举例说明在机器人工程中我们运用动力系统方法到底想得到什么。

假设，一个移动机器人正在某个环境中运动，并且正沿图 4.8 显示的轨道运动。三个描述机器人轨道的参数分别为位置坐标 $x(t)$，$y(t)$ 和方向 $\varphi(t)$。和所有的动力系统一样，这三个变量可以通过微分方程来描述。在这个机器人的例子中三个变量是共轭的，因为机器人控制程序、运动的物理过程和环境的影响意味着 x，y 和 φ 不能完全独立地改变。

在一个真实的机器人中，我们不知道哪些微分方程式描述机器人的运动，但是这里让我们假设，作为论据，式（4.1）定义了一个特定机器人的运动：

$$
\begin{aligned}
\frac{dx}{dt} &= \dot{x} = -(y+\phi) \\
\frac{dy}{dt} &= \dot{y} = x + 0.15y \\
\frac{d\phi}{dt} &= \dot{\phi} = 0.2 + xz - 10z
\end{aligned}
\tag{4.1}
$$

式（4.1）可以求解，例如在数值上使用龙格-库塔（Runge-Kutta）方法，得出如图 4.3 所示函数 $x(t)$，$y(t)$ 和 $\phi(t)$。

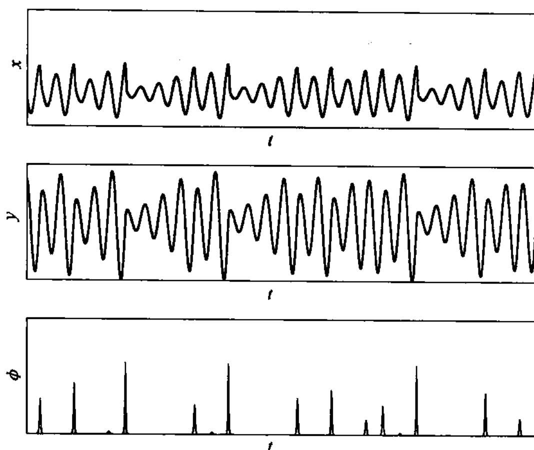


图 4.3　式（4.1）的解

将 x、y、ϕ 用图示表示是有趣的（见图 4.4）。这是被式（4.1）定义的系统相空间，也就是在这个空间下完全定义了系统，而且允许系统预测将来的状态。

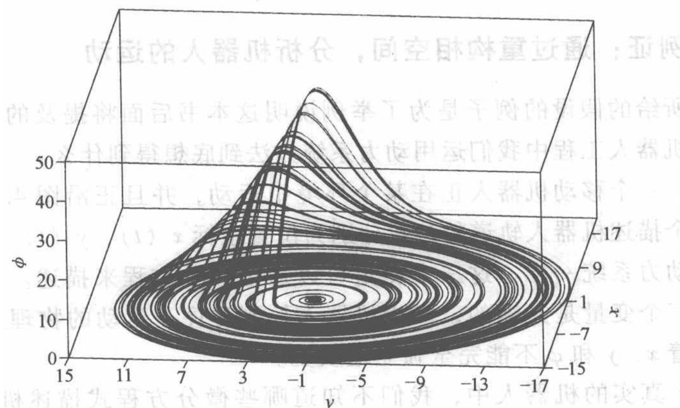


图 4.4　式（4.1）定义的系统相空间

正如前面所说，真实的移动机器人通常是不可能被像式（4.1）定义确定的。然而，通过一个名为时滞嵌入的方法，来重构机器人的相空间是可能的（参见 4.2.3 节）。图 4.5 显示被重构的吸引子，而且给出了图 4.4 中的真实的吸引子的一个近似。

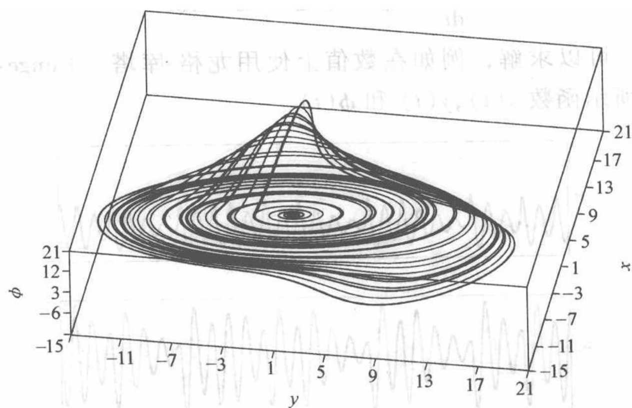


图 4.5　式（4.1）定义的系统的重构相空间

总之，本章介绍的分析机器人—环境交互作用的动力学方法如下：

1）观察机器人在空间（x、y，φ）中随着时间变化的运动。

2）重构这个运动的相空间，使用 4.2.3 节介绍的方法。

3）分析被重构的吸引子的动态特性，比如说敏感的初始状态或吸引子的维

数。

4.2.3 吸引子的重构

分析动力系统的第一步，就是重构它的相空间。理论上，这不会像理想钟摆的情况一样容易。然而，幸运地，相空间能从物理系统观察值的时间序列 $x(t)$ 中，通过运用一种名为时滞嵌入 [Peitgen et al, 1992, Kantz and Schreiber, 1997, Abarbanel, 1996] 的方法被重构。图 4.5 显示了这种方法运用在被式 (4.1) 定义的动力系统的结果。

1. 时滞嵌入：例证

假设我们有了被式 (4.2) 定义的一个动力系统（该例子的完整讨论见 [Kaplan and Glass, 1995] f)：

$$\begin{aligned}\dot{x} &= y \\ \dot{y} &= -bx\end{aligned} \tag{4.2}$$

变量 $x(t)$ 和 $y(t)$ 完全描述了这个系统，它们构成了系统的相空间。

与"机器人行为的观察"方案一致，假设用在时间离散点 t 上记录时间序列 $S(t)=x(t)$ 的方式观察系统行为。为了要重构来自 $S(t)$ 这一系统的相空间，显然需要重构唯一的 $y(t)$，因为 $x(t)$ 是由 $S(t)$ 给出的。

式 (4.2) 指出 $y=\frac{dx}{dt}=\frac{dS}{dt}$。因此，相空间能通过计算 $S(t)$ 的导数重构。

$x(t)$ 的导数由式 (4.3) 给出：

$$\frac{dx}{dt}=\lim_{h\to 0}\frac{x(t+h)-x(t)}{h} \tag{4.3}$$

从式 (4.3) 给出的定义，可以通过式 (4.4) 近似地求出 $S(t)$ 的导数：

$$\frac{dS(t)}{dt}=\dot{S}(t)=\frac{S(t+h)-S(t)}{h} \tag{4.4}$$

由于已经在离散时间点 t 上记录了 S，因此选取合适的 h，应用式 (4.4)，即可估计 $\dot{S}$，从而，重构相空间可绘出 $S(t)$ 对$\frac{S(t+h)-S(t)}{h}$关系曲线。因为所需要的所有信息就包括在 $S(t)$ 和 $S(t+h)$ 中，简单地绘制 $S(t)$ 对 $S(t+h)$ 的图就足够了。

这种方法可以推广到更高的维数，因此可以应用在重构更高维数的吸引子。这在后面会继续论述。

2. 通过时滞嵌入方法重构吸引子

假设测量智能体行为随着时间变换的描述性成分，例如智能体在 (x, y) 空间的运动，得到两个时间序列 x 与 y。吸引子——$D(t_n)$ 通过相空间取得的

轨迹，便可以通过式（4.5）给出的延时嵌入法而被重构：

$$D(t_n) = (x(t_n - (p-1)\tau), x(t_n - (p-2)\tau), \cdots, x(t_n - \tau), x(t_n)) \quad (4.5)$$

这里出现了一组连续的时间序列 x（τ），嵌入维 p，嵌入延迟 τ。

为了通过时滞嵌入的方法从观测的时间序列重构系统的空间，必须对以下两个参量选择其一：嵌入维 p 和嵌入延迟 τ。

1）选择嵌入维，就有三种可能的情况：①嵌入维太小以至于不能重构吸引子；②刚刚好；③太大了。只有第一种情况会产生误差，因为一个吸引子的维数比选择的嵌入维大的话，就不能被完全展开，这就意味着时间上远的点竟变成空间里的邻点（因为这些空间上的邻点是时间上远离的点，所以它们被称为"虚假最近邻点"）。如果嵌入维相等或者稍大，重构显然是没有问题的。如果嵌入维远大于吸引子的维数，理论上也是没有问题的，吸引子能够被完全重构，但是实际操作中的问题使这种情况也不是我们想要的（计算和准确性）。所以最好要选择最小的嵌入维。

一种已有的确定合适的嵌入维的方法是用［Kennel et al，1992］提出的虚假最近邻点法。这种方法可以确定重构相空间中虚假最近邻点的个数（在重构的相空间中接近，但在时间序列中离得很远）。当其个数接近零时，吸引子刚好能展开，而且没有自身的干扰。

2）选择嵌入延迟。第二个时滞嵌入法所选的变量是嵌入延迟 τ。对 τ 的正确选择意味着，与 x（t）相比，所观察的时间序列的样本 x（$t+\tau$）包含新信息的点。举例说，如果一个移动速度慢的系统在一个高采样频率下采样，τ 将变大，因为它会在 x（$t+\tau$）之前采了很多样。另一方面，如果采样频率相对于系统的运动慢的话，τ 将变小。

有一种定性的方法可以看到 τ 逐渐增加的影响。对于小的 τ，x（t）和 x（$t+\tau$）本质上相等。如果把它们绘制成图，那么所有的点落会在一条对角线上。当 τ 变大，那么重构的吸引子将会从该线上扩展开来。这种扩展可以指导我们作出关于 τ 的适当的选择［Rosenstein et al，1994］。

有两种进一步确定该点的方法，在这些点上 x（t）与 x（$t+\tau$）包含不同的信息。［Kaplan and Glass，1995］中提出，当 x（t）与 x（$t+\tau$）的自相关数低于 $e^{-1}=0.37$ 时，就得到一个合适的 τ。［Fraser and Swinney，1986］认为：当被认为是自相关函数［式（4.6）］概括的互信息[⊖]取其第一个极小值时，得到样本中包含新信息的点：

$$MI = H(x) + H(x+\tau) - H(x, x+\tau) \quad (4.6)$$

H（x），H（$x+\tau$）与 H（x，$x+\tau$）见式（3.52）与式（3.54）的定义。

⊖ 互信息是信息论里一种有用的信息度量，它是指两个事件集合之间的相关性——译者注。

现在有了合适的工具实现相空间重构，下面我们将首先看一个实际的机器人工程例子。

4.2.4 重构吸引子：机器人学的例子（避障）

有一台沿 x 轴运动的避障机器人，通过其头上一架照相机每隔 250ms 记录机器人的位置，获得 26 000 个数据点，图 4.13（右图）列出了这些数据点的一部分。

为了重构吸引子，利用式（4.5）确定一个适当的嵌入延迟 τ 和嵌入维 p。

我们将依据实际的考量来选择嵌入维 p。如前所述，本实验中所用的机器人是差动式机器人，有 2~3 个自由度，所以 $p=5$ 很可能就是嵌入维的一个很好的选择。

为了确定嵌入延迟 τ，定义一点，其 $x(\tau)$ 的自相关函数降到 $e^{-1}=0.37$ [Kaplan and Glass，1995 年]，或在该点的互信息 *MI*（见式 4.6）有其第一个极小值 [Fraser and Swinney，1986 年]。

图 4.6 显示了 $x(t)$ 的自相关函数和互信息。对于 $\tau=29$，自相关函数下降到小于 0.37，并且互信息也取得了最小值。因此利用式（4.5），选择 $\tau=29$ 和 $p=5$[⊖] 来重构吸引子。

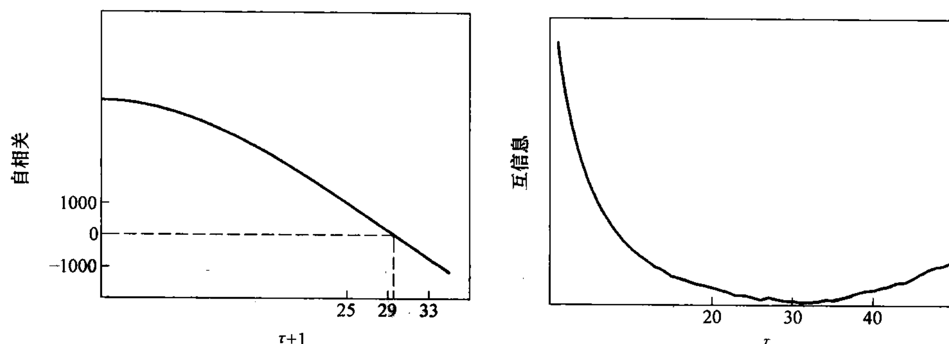


图 4.6 对于避障行为 $x(t)$ 的自相关函数（左图）和互信息（右图）的图像
（当 $\tau\approx29$ 时自相关系数下降到小于 0.37，并且互信息取得了第一个极小值）

最后，当重构了吸引子，我们要检验一下吸引子是否被有效地“触发”（详细的论述参见后文）。图 4.7 展示了 $p=3$ 时的吸引子（因为更高维数不能形象地演示出来），并且即使在这么低的维数下，吸引子仍然能很好地展开。为了分析避障机器人的相空间，我们将用更高的维数，基于以上讨论的实例，计算李雅普诺夫指数或者相关维数，如图 4.7 所示。

⊖ $p=5$ 是一个折中的、安全的选择——参见前文关于移动机器人自由度的相关讨论。

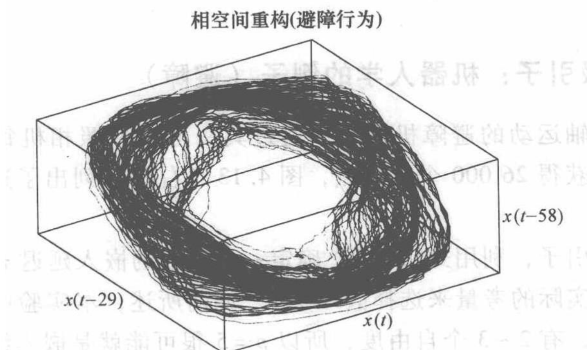


图 4.7 避障机器人的相空间的三维重构

4.2.5 实验设置和行为定量描述的应用

在 2.4.2 节介绍过，假定存在某一关于机器人行为（来自机器人、任务和环境的相互作用）的定量描述子，如果让其中两个量不变，有规则地改变第三个量，并用该定量描述子来描述改变的那个量和机器人行为之间的关系，就可以用这个描述子来准确描述机器人、任务和环境这三者中的一个量。

图 4.8 就显示了这样的一个实验设置。在图 4.9 所示的一块场地中运行一个麦哲伦专业版移动机器人，通过每次改变机器人、任务和环境这三者中的一个，我们做三个实验。

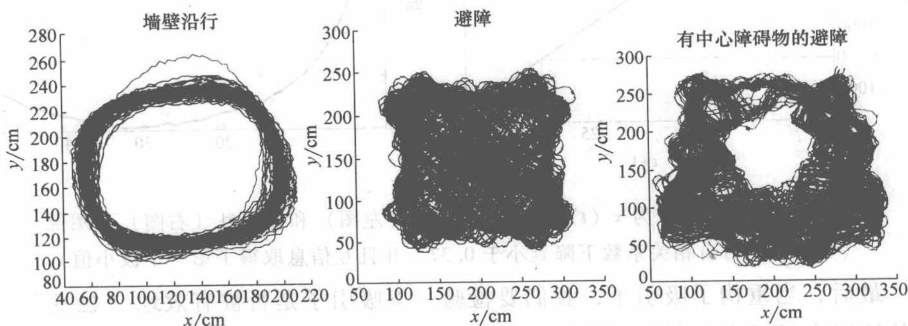


图 4.8 用于本章的三个不同的数据集（机器人轨迹）［1406（墙壁沿行）、2406 集（台球式避碍）和 0507（有中心障碍物的台球式避碍）。从 1406 集到装置 2406 集，机器人和环境保持不变，但任务发生改变。1406 集和 0507 集的机器人和任务相同，但环境发生改变。］

在实验Ⅰ和Ⅱ（图4.8的左图和中间的图）中，保持机器人和环境为常量，改变任务。在实验Ⅱ和Ⅲ（图4.8中间的图和右图）中，保持机器人和任务为常量，改变环境这个成分。这三个实验的目的是测量改变的量对机器人行为的影响。

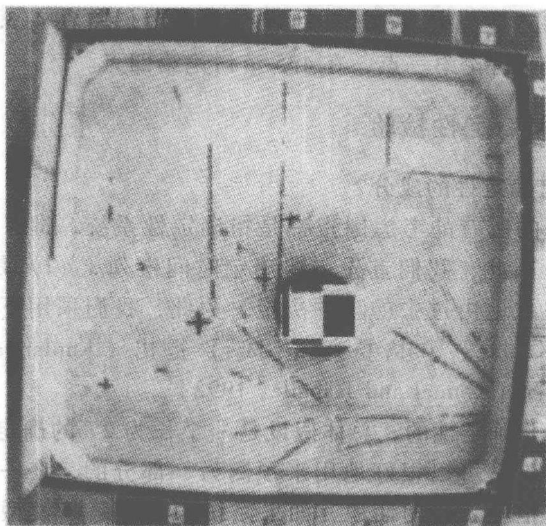


图4.9 一个简单的机器人场地

4.3 通过相空间分析量化描述（机器人）行为

4.3.1 确定性混沌

要预测在圆形轨道上恒速运行的火车模型的运动，不会有多少困难，只有一个非常小的预测误差。要预测掷骰子的结果时，我们可以做一个随意的猜测，或者总是预测出骰子上的所有数字的平均数。前面的系统是确定性的、完全可以预测的，后面的系统是随机的、不可预测的。

还有第三种系统：它是确定的而不是任意的，但是不可预测的，除非预测涉及邻近的、短期的未来。天气和台球的运动轨迹属于这一类系统，移动机器人在一些情况下的运动也属于这一类系统。这种系统展示了确定性的混沌。说它是确定的，是因为这种系统的运动是由确定的规律，如运动定律所控制的，而不是任意的。说它是混沌的，是因为它似乎是一个随机的系统，并不总是能被预测。

具体来说，如果系统在研究中表现出以下四个特性就是所谓的确定性混沌：

1）系统的行为是（主要的）确定性的。

2）系统的行为是有界的、稳定的。

3）系统的行为对初始条件的微小变化十分敏感。

4）系统的行为是非周期性的。

前面的两点可以决定提出的方法是否适合于我们现有的信号种类，后面的两点决定了这种信号是否表现出确定性的混沌。我们将在4.3.2节讨论第一点和第二点，在4.4节讨论第三点，在4.5节讨论第四点。

4.3.2 确定性和稳定性检验

1. 系统是否有确定性的成分？

这一章中提到的所有的考虑因素都是指确定性系统，即，主要不是由随机行为控制的系统。因此，我们首先需要确定时间序列 $x(t)$ 是否是确定性的，也就是说是不是任意地由过去的事件决定。为此，我们采用下面的方法，这个方法由开普兰（Kaplan）和格拉斯（Glass）提出［Kaplan and Glass，1995，p. 324ff］（也可以看［Kennel and Isabelle，1992］。

判断信号是否是确定性的，总体假设是一个长为 $2T$ 的确定性信号 D，这个信号的前半部分应该是可以很好地用来预测另一部分的，在一个完全随机的系统里，这个假设被拒绝。换句话说，如果系统的一个基于模型的预测是完美的（没有丝毫误差的），那么这个系统是完全确定的。如果有一些小的预测误差，这个系统会有一个确定的成分，而如果一个基于模型的预测只是跟一个随机的猜测一样，那么这个系统就是完全不确定的。

为了说明信号 D 的前半部分是不是后半部分的一个很好的预测，我们把时间序列 D 分成两个长为 T 的部分，并像式（4.7）那样构建一个嵌入的 D：

$$D(T+i) = [D(T+i), D(T+i-1), D(T=i-2)], \forall i = 3, \cdots, T \tag{4.7}$$

换句话说，我们用一个时间延迟 $\tau=1$ 和嵌入维 $p=3$ 为时间序列的后半部分建立一个嵌入序列。（当然 τ 和 p 也可取其他值。）

为了预测 $D(t_k+1)(T<t_k<2T), D(t_k+1)p$，我们测定在欧几里得（Euclidean）距离上距 $D(t_k)$ 最近的点 $D_c(t_c)(0<t_c\leqslant T)$，并用 $D(t_c+1)$ 预测 $D(t_k+1)$。按这种方式，后半部分所有的点都被预测出来了。我们总是只往前作一步预测，如图4.10所示。

然后，我们计算一下均方预测误差 ϵ。为了判断这个误差是“大”还是“小”，我们把它和简单地用信号前一半的均值来预测后一半的基线误差 ϵ_b 进行比较。在一个完全随机的信号里，比值 ϵ/ϵ_b 是1或者大于1，说明平均数可能是最好的预测，因此这个系统是不确定的。另一方面，如果比值 ϵ/ϵ_b 小于1，这就表明时间序列的前半部分确实是后半部分的一个好的预测，因此，时间序列有一个确定的成分。

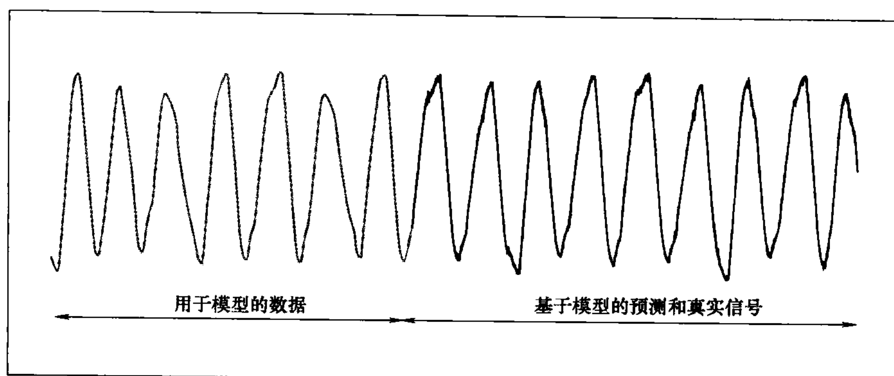


图 4.10　机器人避障行为中沿 X 轴运动的预测

另外，还有第二种判断时间序列是确定的（即信号值是由之前的信号值决定的），还是随机的（即信号值与之前的信号值独立）的方法。

通过简单的绘制 $x(t)$ 和 $x(t-\tau)$，可以从视觉上看出之前的和现在的信号值是否有因果联系。这些绘图称为回归图，图 4.11 中展示了三个例子。

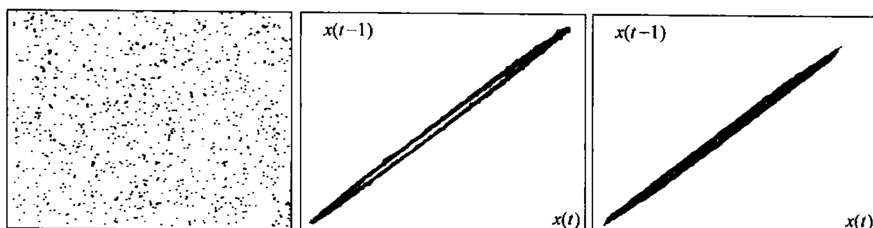


图 4.11　随机噪声的回归图（左图），墙壁沿行行为（中图），避障行为（右图）
（可以清楚地看到，墙壁沿行和避障都有一个确定的成分）

2. 稳定性检验

对一个动力系统（比如与环境相互作用的机器人）的分析，是为了找到和量化这个相互作用的统计特征。这就是说，如果这个统计特征是随时间变化的、不稳定的，本书中提到的时间序列分析方法就不能应用。因此，在用本书总结出的方法来分析一个信号之前，必须确定问题中的信号是稳定的。

如果一个信号在它的整个时期里显示出相似的行为，就把它定义为是稳定的。这里“相似的行为”定义为在整个时间序列里有相似的均值、标准差和自相关结构 [Kaplan and Glass, 1995, p. 314] [Pena et al, 2001, p. 29]。实际上，真实世界的信号通常都显示不变的均值和自相关结构，但是在整个时间序列里有不同的变化。这种信号有时会涉及到“弱稳定性”，但是考虑到它们足够接近理想的稳定性，可以当做是稳定信号。稳定性的概念很明显是相对于数据中出现频率最高的部分，这意味着只有所用的数据包含一个最长时期稳定的量，

这个分析才是有意义的。

为了检验稳定性，必须检验数据的不同部分的均值和标准差是否存在显著差异（在下面的检验中我们不考虑自相关结构）。

研究这个问题的一个简单检验就是将时间序列分成许多部分，检验每个部分的总体分布与其他的是否不同。由于总体分布并不知道是否是正态的（这是可以检验的，可是要用3.3.2节所描述的方法），适合用一个像3.4.4节中介绍的非参数方差分析。如果总体分布是正态的，可以用一个t-检验来代替非参数方差分析。

3. 非参数游程⊖检验

另一个检验稳定性的方法是这样的：像之前那样将数据分段（对于下面描述的这个检验，至少需要41段）。然后，我们确定每一段的均值是在整个时间序列的中值之上还是之下。这会得出一个序列的结果，就是说，41个值像“AABABBABBAA…”这样排列下去，其中“A”表示这一段的均值在整个时间序列的中值之上，“B”表示在它之下。

如果时间序列是不稳定的，例如由于数据有一个线性上升的趋势，我们将得到一个有非常少游程的序列（参见3.5节对“游程”的定义），因为所有的“先前的”的数据将趋于一个低于中值的均值，而所有后面的数据将趋于在中值之上。另一方面，如果数据是稳定的，我们预期“As”和“Bs”的分布是随机的。换句话说：太少或是太多的游程表明观测值间有依赖性，因而是不稳定的。因此，我们在问题中对一个随机的“As”和“Bs”分布进行（像在3.5节介绍的那样）检验。如果分布是随机的，那么就认为数据是稳定的。

给数据分段要注意，比如说是一个纯正弦波，显然它是稳定的。可是下面这种情况是可能的，毫无技巧地给一个正弦波分段，以至于得到一个像“AB-ABAB…”这样的序列，这显然不是随机的。因此，对我们步骤的明确表述应该是：如果时间序列中至少有一段，在整个游程检验中是随机的，那么可以认为这个时间序列是稳定的。

4. 使不稳定的数据稳定

不稳定的信号往往可以通过简单的转换变为稳定的。最简单的一个例子就是计算时间序列序列值的一阶差分，如果一阶差分仍不稳定，就重复这个过程（二阶差分）。

另一个显然需要的步骤就是去除线性趋势。这可以通过简单地减去与时间序列拟合最好的线性函数 $y = ax + b$ 来得到（线性回归）（这在3.6.1节已介绍过）。

⊖ 非参数游程检验也称为“连贯检验”，是根据样本标志表现排列所形成的游程的多少进行判断的检验方法。连续出现样本标志（比如，男或女、左或右的区段称为游程——译者注。

如果信号 x（t）显示随时间指数性增长，有时可以通过使用 $x(t)/x(t-1)$ 分析来使它稳定［Kaplan and Glass，1995，p. 315］。其他的可以使不稳定信号稳定的转换有对数和平方根转换。例如：一个指数信号，可以通过取对数线性化，再通过计算一阶差分使其稳定。类似的，遵循能量定律的时间序列可以通过计算平方根或是取更高次的方根使其直线化。

在确定了智能体行为的描述子（记录的时间序列）的确主要是确定的、稳定的之后，我们现在可以量化分析系统的相空间。

4.4　初始条件敏感性：李雅普诺夫指数

考虑下面一个思想实验：你观察一个物理系统，如钟摆在物理空间的运动。这个运动相当于一个相空间的运动“轨道”。在钟摆的这个例子中，轨道是椭圆形的。

如果你想象钟摆从某个位置 ϕ 出发，与此同时（这是个思想实验）在点 $\phi+\Delta$，Δ 是一个小距离，那么我们通过物理空间和相空间得到两个开始时彼此离得很近的运动。对于有些系统，如钟摆，这两个运动随时间既不会变远也不会变近；对于其他系统，两个轨道会收敛成一个轨道，还会发散并且迅速地远离（这是可能发生的，例如台球的轨迹）。两个开始时彼此无穷接近的轨道发散或收敛的速度描述了一个引人注目的特性——它称为李雅普诺夫指数。

一个混沌系统最鲜明的特点之一就是对该系统中变量的变动敏感性：在相空间中两轨迹开始相互靠近，随着时间的推移，又将互相偏离，系统越混乱，发散度越高。

考虑一个确定动力系统的某个 S_o 态以及与之相应的在相空间中的位置。随着时间推移，这种系统状态在相空间中的变化沿着一个确定的轨迹。让系统的另一个 S_1 态沿另一个不同的完全确定的轨迹任意靠近 S_o 态。如果 d_o 是两种状态在相空间中时间为 0 时的初始分离度，则它们在 ts 后的分离度 d_t 可以用式（4.8）来表示

$$d_t = d_o e^{\lambda t} \tag{4.8}$$

或者换一种说法，考虑初始误差 E_0 的平均对数增长（距离为 $|x_0-(x_0+\in)$，$\in$ 是一个任意小的值，x_0 是相空间内一点），如果 E_k 第 k 步的误差，E_{k-1} 是前一步的误差，则平均对数误差增长可用式（4.9）表示

$$\lambda = \lim_{n\to\infty}\lim_{E_0\to 0}\frac{1}{n}\sum_{k=1}^{n}\ln\left|\frac{E_k}{E_{k-1}}\right| \tag{4.9}$$

λ（单位为 s^{-1} 或者 bit/s，取决于用自然对数或者用以 2 为底的对数）称为李雅普诺夫指数。

对于一个 m 维相空间，有 m 个 λ 值，即每一个维度有一个 λ 值。如果任何一个或多个分量是正的，那么附近状态的轨迹在相空间和系统中成指数式发散的，就可以认为是混沌的。因为任何一个系统状态的变量都是不确定的，所以如果混沌出现的话，那么我们很快就不知道系统处于什么样的状态了。正的李雅普诺夫指数越大、关于系统的信息就丢失越快。只能知道系统的状态处于相空间中的某一个轨道中，例如在吸引子的某处。

李雅普诺夫指数是最有用的量化描述混沌的手段之一，因为它直接反映了系统是否真正是混沌的，并且可以计量混沌程度。同时，李雅普诺夫指数成为对未来状态预测分析必需的知识。

4.4.1 时间序列李雅普诺夫指数的估计

一个确定一个吸引子的李雅普诺夫指数来描述一个物理系统行为的方法，就是通过观察系统运动的时间序列来估计它 [Peitgen et al, 1992]。然而，对一个时间序列的李雅普诺夫指数的估计是很重要的，并且通常非常依赖于参数设置。因此，简单采取现有的软件包，选取似乎是恰当的参数设置或者计算相关指数都是不够的。相反，应计算使李雅普诺夫指数不变的设置范围，称为标度区间。这些标度区间显示出良好的参数设置和可靠的结果。

有许多软件包可用于从时间序列中计算李雅普诺夫指数，例如 [Kantz and Schreiber, 1997] 中介绍的 [Kantz and Schreiber, 2003]、 [Abarbanel, 1996] 中介绍的 [ANS, 2003] 和基于 [Wolf et al, 1995] 的 [Wolf, 2003]。在这里显示的结果是用沃尔夫（Wolf）的软件获得的。

1. 用沃尔夫程序估计李雅普诺夫指数：机器人学实例

我们非常有兴趣计算两个不同机器人行为的李雅普诺夫指数：墙壁沿行和避障。图 4.12 显示了先锋Ⅱ号机器人执行这些行为的轨迹（由一个在顶部的照相机获得）。

为了进行数据分析，我们单独地观察每个行为在 x 和 y 轴上的分量，并估计它们的李雅普诺夫指数。图 4.13 显示了机器人两种行为在 x 方向的移动分量。

根据图 4.12 所示轨迹，我们推测墙壁沿行行为的可预测性远远强于避障行为的可预测性，并且因此有一个更小的李雅普诺夫指数。我们用 [Wolf, 2003] 来研究。

为了应用沃尔夫程序，以下参数需要具体给定：

1）通过时滞嵌入为潜在吸引子的重构嵌入延迟 τ（参见 4.2.3 节）。[Kaplan and Glass, 1995] 提出时间序列的自相关作用降低到 $1/e=0.37$ 以下时可以找到一个合适的 τ 值。另外，可以使用互信息 [Fraser and Swinney, 1986]（后文将继续讨论）。一般规则为：τ 应选择这种重构的吸引子是完全展开的（即不

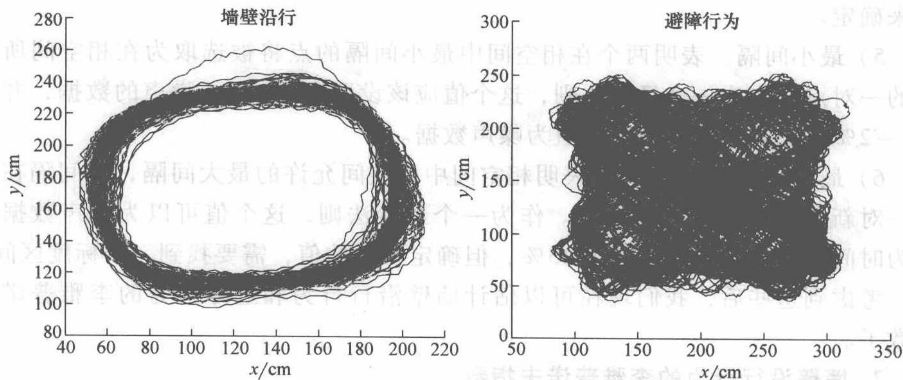


图4.12　一个先锋Ⅱ号移动机器人墙壁沿行（数据集1406）和避障行为（数据集2406），（由一个在顶部的照相机获得）

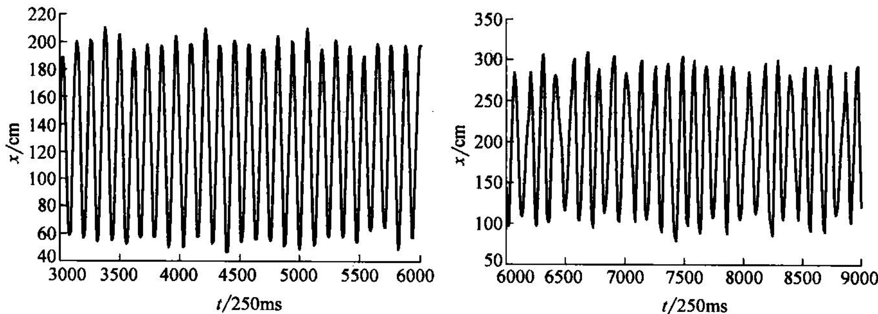


图4.13　墙壁沿行（左图）和避障（右图）行为在x轴的移动分量

包含自我影响），并显示出一个定义较好的结构。

2）嵌入维数p。泰肯斯（Takens）和麦恩（Mäne）的理论［Mäne，1981］［Takens，1981］说明了如果嵌入维数p满足$p=2d+1$，其中d为吸引子的维数，则一个吸引子通过时滞嵌入可以完美地重构（参见4.5节中关于一个吸引子维数的问题）。然而在实践中，较低的嵌入维数通常是足够的，对于大多数移动机器人系统的一个3~5维嵌入维数就足够了。p的选择受限于可用数据量：作一个粗略估计，用于分析所需要的数据点数为30^d［Wolf，2003］，其中d是吸引子的维数（见4.5节）。

一个用来定义恰当嵌入维数完善的方法就是伪最临近点法。这种方法在［Kennel et al，1992］中有介绍。

3）时间步长。这是用于获取时间序列的简单采样率（例如在顶部的照相机每250ms获得一个机器人的位置坐标）。

4）时间演化$evolv$。这个参数确定了一对点通过相空间有多少步骤，用来估计它们随时间发散的情况。正确的设置（通常3~12）必须通过找到一个标度区

间来确定。

5）最小间隔。表明两个在相空间中最小间隔的点将被选取为在相空间所跟踪的一对点。作为一个经验法则，这个值应该设置为0，为无噪声的数据，并以1% ~2%的范围内的时间序列值为噪声数据。

6）最大间隔 *maxdist*。这表明相空间中两点间允许的最大间隔，该间隔在找到一对新点来估计发散度之前。作为一个经验法则，这个值可以为噪声数据设置为时间序列值范围的10% ~15%，但确定合适的值，需要找到一个标度区间。

考虑到这些后，我们现在可以估计墙壁沿行行为和避障行为的李雅普诺夫指数了。

2. 墙壁沿行行为的李雅普诺夫指数

对于墙壁沿行行为，由13000个数据点构成（54min的机器人操作），在 $\tau=30$ 和最小间隔为2（x 轴时间序列范围是164）的情形下，我们取得的结果如表4.1所示。

表4.1 应用沃尔夫程序 [Wolf, 2003] 的不同参数设置下的墙壁沿行行为李雅普诺夫指数估计

evolv	*maxdist*	$\lambda(p=3)$	$\lambda(p=4)$	$\lambda(p=5)$
2	20	0.005	0.004	0.003
3	20	0.006	0.02	0.03
4	20	0.03	0.03	0.03
5	20	0.03	0.03	0.03
6	20	0.04	0.03	0.03
7	20	0.04	0.04	0.03
8	20	0.03	0.04	0.03
9	20	0.04	0.03	0.03
10	20	0.04	0.03	0.03
6	15		0.05	
6	20	0.04	0.03	0.03
6	25		0.02	
6	30		0.02	
6	35		0.02	

结果从始至终非常统一，同时选取演变时间为6和嵌入维数为4，我们估计得到 $0.02\text{bit/s}<\lambda_{wf}<0.03\text{bit/s}$。

3. 避障行为的李雅普诺夫指数

对于避障行为（26000个数据点，运行108min，范围239），当 $\tau=29$（见图4.6），最小间隔为2，我们得到表4.2所示的结果。

可以看到一个标度区间为 $3<evolv<10$，因此选择 $evolv=6$。最大间隔没有表现这样一个明显的标度区间，但是值在25 ~30的结果相当一致。当 $p=4$ 和 $p=5$ 时，结果很一致，我们取 $p=4$。由此，我们估计得到 $0.11\text{bit/s}<\lambda_{oa}<0.13\text{bit/s}$。

就像之前假设的，避障行为比沿墙行为更混沌，李雅普诺夫指数更高。

表 4.2　根据不同参数设定，应用沃尔夫程序 [Wolf, 2003] 得到的避障行为的李雅普诺夫指数的估计值

evolv	*maxdist*	$\lambda(p=3)$	$\lambda(p=4)$	$\lambda(p=5)$
2	28	0.02	0.02	0.01
3	28	0.14	0.12	0.10
4	28	0.15	0.12	0.11
5	28	0.15	0.12	0.11
6	28	0.15	0.12	0.11
7	28	0.14	0.12	0.10
8	28	0.15	0.12	0.10
9	28	0.14	0.12	0.10
10	28	0.14	0.12	0.10
6	20	0.18	0.14	0.10
6	25	0.16	0.13	0.11
6	28	0.15	0.12	0.11
6	30	0.14	0.11	0.10
6	35	0.13	0.11	0.10
6	40	0.11	0.10	0.09

4.4.2　通过信息损失和预测时域预测估计李雅普诺夫指数

1. 信息损失

当李雅普诺夫指数通过单位时间的信息损失表示时，可以通过确定 t_0 时刻系统有多少有用信息，和增加时间 t 减少多少信息进行估计。这种方法可以通过以下途径实现。

假设我们观察一个机器人及其周围环境，记它的位置为 $x(t)$ 和 $y(t)$。机器人在此位置的信息 $I(t)$（单位为 bit）通过式（4.10）给出

$$I(t) = \ln B(t) / \ln 2 \tag{4.10}$$

式中，$B(t)$ 是 t 时刻我们已知的 x、y 轴上清楚位置的个数。例如，我们针对机器人在 x 轴的 256 个清楚位置上的一个位置（比如说一台相机的 256 个像素），有 8bit 的可用信息。

我们用这种思想实验确定的机器人的位置是根据以前的数据预测的，就像 4.3.2 节中描述的方法一样。机器人的初始位置是完全已知的，而且仅仅由我们照相机的分辨率决定。

假设用一部顶部照相机测定机器人的位置。同时，假设在 $t=0$ 时刻，知道机器人的位置，精度为 1 像素。因为在 $t=0$ 时刻，我们能够在精度为 $1/R$ 像素内指定机器人的位置，为了估计 $t=0$ 时刻的信息量，我们仅采用范围 R，定义

$$R = x_{max} - x_{min}$$

式中，x_{max}和x_{min}是观察的最大和最小的x位置，而且估计I（$t=0$）时，取$B=1$［式（4.10）］。

随着时间的推移，由于预测误差的增加，我们越来越难预测机器人的位置。例如，在t_1时刻的一个预测误差1，意味着我们在x轴只能指定精度为$2/R$像素的机器人的位置，已确定的像素位置±1像素。

我们一开始在t_0时刻有256个清楚的位置。这意味着$I_0=8\text{bit}$。对于在t_1时刻的预测误差1，我们可以将机器人定位于128个确切的位置（7bit）。也就是说，要减少1bit的信息需要t_1-t_0的时间，我们可以估计李雅普诺夫指数为$\lambda \approx \frac{1\text{bit}}{(t_1-t_0)\ s}$。

2. 预测时域

李雅普诺夫指数单位为bit/s，很长时间以来作为一个预测的信号，说明信息损失取决于信号的混沌特性。一个完全的无噪声、无混沌的信号，其李雅普诺夫指数为0，可以预测任意长时间，而且没有预测误差。另一方面，一个混沌信号不能预测任意长时间，因为每一步预测不确定性都会增加，直到最后预测结果不再比一个根据事实或经验作出的猜测更好。此时，就出现了完全损失可预测性的情况。

如果，10min前要求你预测户外的温度，并对你的预测给出一个不确定性的说明。因为只有一个很小的不确定间隔，你可以简单地认为温度保持此刻的温度不变，做个很好的预测。如果2h前要预测温度，会更不确定，甚至是12h前……

当可预测性完全失去时，就可以用李雅普诺夫指数来计算，例如，当数据的任何模型都不比一个有根据的猜测更好时（我们把这叫做“预测时域”）。需要牢记的是李雅普诺夫指数是一种平均的测量方法，也可能出现这样的情况，预测会比根据事实或经验作出的猜测好而且往往会超过估计预测时域很多，但平均起来，只有当可预测性的完全损失将出现时，才用李雅普诺夫指数来估计预测时域。

由例证可知，假设你正在测量一些工业设备的压力，你会想要预测过一段时间压力会怎么变化。你在过去很长一段时间里已经做过一系列的压力测量工作，估计李雅普诺夫指数是0.5bit/s。使用的压力传感器能测出256种不同的压力值，也就是说$\log_2 256/\log_2 2=8\text{bit}$。这意味着一般来说在16s后，可预测性损失会出现。换句话说：平均起来，在16s后即使是一个“黄金标准”的压力分布模型也比不上对其进行根据事实或经验作出的猜测。

“根据事实或经验作出的猜测”在这里的意思是不依靠过去值的一个具体的情况对一个值进行预测，而是利用该信号的全局性质。这就是比较我们模型预

测表现的基线（这的确考虑到了先前值）。对每一个点 x（t_p）应该使用一条简单的基线，该点随着时间的发展，我们会预测从时间序列里随机选择其他点 x（t_m），然后将 x（t_m）的后继点作为 x（t_p）后继点的预测。

正因为李雅普诺夫指数能用来估计这个时间点，在该点，根据事实或经验作出的猜测，也会产生和“黄金标准”模型一样的小的预测误差（平均而言），我们也一样能逆向推导：确定那个时间上我们或许也会作出对信号的随机猜测的点，然后再从这点推导李雅普诺夫指数。

我们将用到的“黄金标准”模型其实就是数据本身。将数据分割为相等的长度为 T 的两半，则我们用第一半的数据作为第二半的模型。这是一个敏感的步骤，因为我们这里只处理确定性的数据，这意味着过去的数据点在一定程度上是未来数据点的预测。

为了预测我们数据的第二部分未来的数据点 D（t_2），$t_2=T+1,\cdots,2T$，我们构建一个三维嵌入 $\boldsymbol{D}$（t_2）如式（4.11）给出那样，再在第一半数据中搜寻向量 $\boldsymbol{D}$（t_1），$1\leqslant t_1\leqslant T$ 是最接近 $\boldsymbol{D}$（t_2）的（欧几里德距离）：

$$\boldsymbol{D}(t_2)=[D(t_2),D(t_2-\tau),D(t_2-2\tau)] \tag{4.11}$$

式中，τ 是指嵌入延迟。

这是一个相空间里的三维重构，对于我们来说已经很熟悉了。

然后，我们再估计后面 k 个数据点 D_m（t_2+1），$\cdots$，D_m（t_2+k），如式（4.12）所示：

$$D_m(t_2+i)=D(t_1+i) \quad i=1,\cdots,k,1<t_1<T,T<t_2<2T \tag{4.12}$$

我们会针对基线对这个估计结果进行比较，这说明我们选择的 D_B（t_r）点是随机的（其中 $1<t_r<T$），然后用式（4.13）估计 D_B（t_2+i）：

$$D_B(t_2+i)=D_B(t_r+i) \quad i=1,\cdots,k,1<t_r<T,T<t_2<2T \tag{4.13}$$

平均模型误差$\overline{D_M-D}$和在平均基线误差$\overline{D_B-D}$一样的点就是预测时域。

4.4.3 用信息损失和预测时域估计李雅普诺夫指数：标准算例

在运用以上所说的两种方法来解决机器人学实例之前，我们会用已知混沌特性的公式来检验。具体来说，我们估计二次方程迭代器和洛仑兹（Lorenz）吸引子这两个很容易理解的系统的李雅普诺夫指数。这用来确认该方法对已知的混沌系统有效。

1. 估计二次方程迭代器的李雅普诺夫指数

二次方程迭代器（也称为逻辑映射）是我们熟知的混沌动力系统，如式（4.14）所示：

$$x(t)=4x(t-1)[1-x(t-1)] \tag{4.14}$$

图4.14给出了一种典型二次方程迭代器的时间序列，图4.15给出了二次方

程迭代器计算后的信息损失，预测的是未来更长一段不断增加时间的情况。在第 1 秒和第 7 秒的信息损失之间是一个线性区域，在这个区域内，6bit 的信息都会丢失。这意味着对于二次方程迭代器来说 $\lambda = 1\text{bit/s}$。

对于预测时域的估计在图 4.16 中已经给出。这表明预测时域所有信息全部丢失的那一点——会在 10s 后达到。由于在开始有 10bit 的信息是可用的（见图 4.15），这说明李雅普诺夫指数为 $\lambda = 1\text{bit/s}$，这和用信息损失法得到的结果是一样的。巧合的是，正确的二次方程迭代器的李雅普诺夫指数的确是 $\lambda = 1\text{bit/s}$ [Peitgen et al, 1992]，因此我们的结果是有序的。

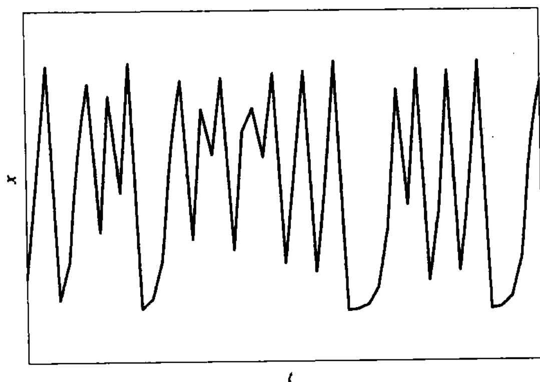


图 4.14 一种典型二次方程迭代器的时间序列

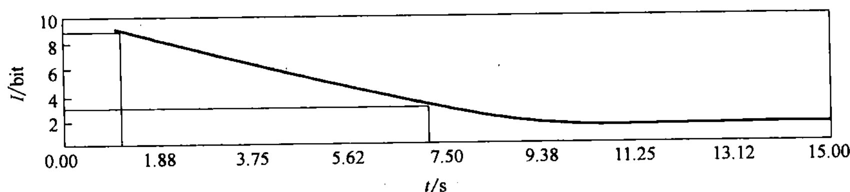


图 4.15 二次方程迭代器的信息损失

（在第 1s 和第 7s 间的可见线性区域，中间 6s 有 6bit 信息损失）

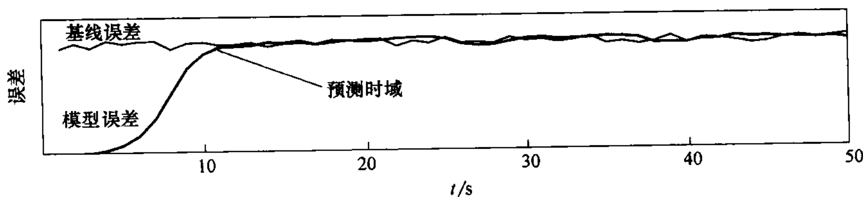


图 4.16 图 4.14 中给出的二次方程迭代器的预测时域

2. 估计洛仑兹吸引子的李雅普诺夫指数

洛仑兹吸引子是另一种我们熟悉的有混沌特性的动力学系统，它是由式

(4.15) 的微分方程来定义的。

$$
\begin{aligned}
\dot{x} &= 16x + 16y \\
\dot{y} &= x(45.92 - z) - y \\
\dot{z} &= -4z + xy
\end{aligned}
\tag{4.15}
$$

图 4.17 给出了式 (4.15) 中 x (t) 的解法，我们就是用这个时间序列来估计 λ_{Lorenz} 的。

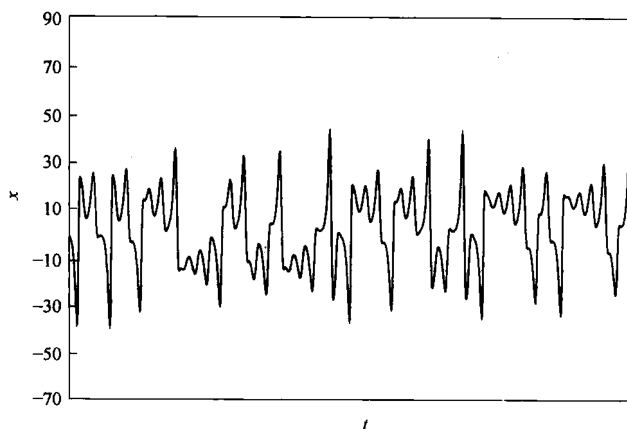


图 4.17　式 (4.15) 中 x (t) 的解法

图 4.18 给出了洛仑兹吸引子在 x 坐标轴上的信息损失和预测时域的估计值。

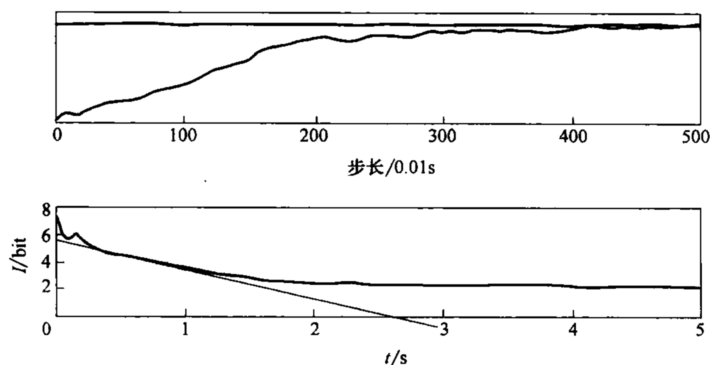


图 4.18　在洛仑兹吸引子在 x 坐标轴 (见图 4.17) 上的信息损失和预测时域的估计值

预测时域大概有 400 多步，这要归功于用龙格-库塔法来解式 (4.15)，使它等于4s。又可用初始信息大约为7.5bit，因此我们用预测时域来估计 $\lambda_{Lorenz} \approx$

$\frac{7.5\text{bit}}{4\text{s}}=1.9\text{bit/s}$。

信息损失（图4.18下部）给出了一个线性区域，这从图中可以看出。这条线表明在2.6s内可用信息从5.6bit减少到0bit，得到一个估计的李雅普诺夫指数为 $\lambda\approx\frac{5.6}{2.6}\text{bit/s}=2.15\text{bit/s}$。

因此，我们估计 $1.9\text{bit/s}<\lambda_{Lorenz}<2.15\text{bit/s}$。精确值为 $\lambda_{Lorenz}=2.16\text{bit/s}$ [Wolf et al，1995]，这和我们的结果相符。

4.4.4 用信息损失和预测时域估计李雅普诺夫指数：机器人学实例

墙壁沿行

图4.19表示机器人墙壁沿行行为的预测时域和信息损失。

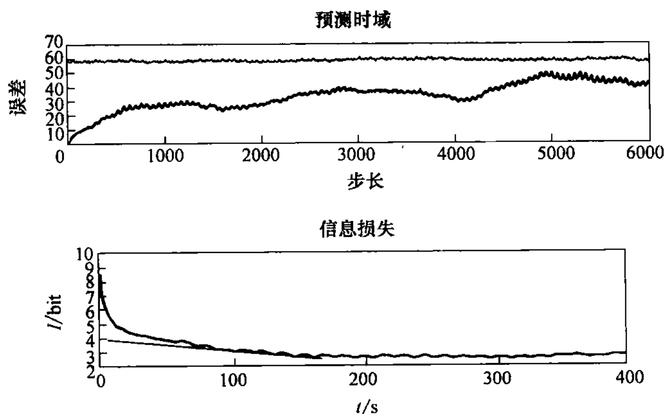


图4.19 墙壁沿行行为的预测误差和信息损失（数据组1406）

从图4.19我们可以看出，墙壁沿行行为（预测度很高）的情况中预测时域线很长。墙壁沿行可重复和可预测性很强，以至于在对机器人未来6000步进行预测时（在4Hz的采样率下多于25min），我们用它以前的动作作为模型来预测其将来的位置仍比随机地猜测更精确。由此可见，李雅普诺夫指数约等于0。

图4.19同样表明了信息损失只在一个有限的区域 t 内呈线性关系，说明我们在有限的一段时间里仅仅能得到指数的信息损失，就像式（4.8）假设的那样。

如果任意信息损失的线性区域能在图中被完全辨识（图4.19下部），这说明在200s里有4bit的信息损失，并得出 $\lambda_{wf}\approx0.02\text{bit/s}$，也就是说基本上等于0。这些结果与用沃尔夫法得出的结果相符（见表4.1）。

1. 避障

我们可以将同样的方法应用于避障行为，如图 4.12 所示。跟以前一样，我们将数据分割成两半，用一半去推测另一半，像式（4.11）那样重构一个三维的相空间。结果已经在图 4.20 中给出。

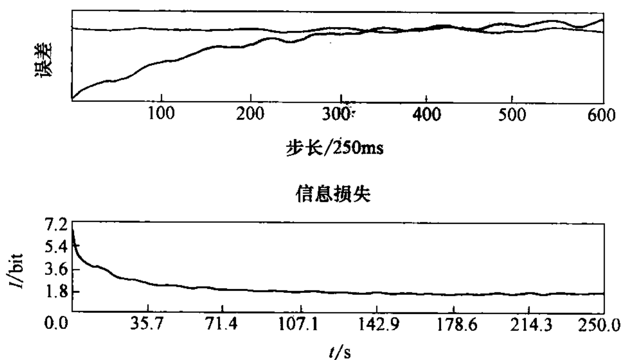


图 4.20　避障行为的预测误差和信息损失（数据组 2406）

图 4.20 表明，不像墙壁沿行那种高可预测性的动作，避障的预测时域会短得多，大约 320 步（80s）。换句话说，在 80s 达到预测时域后，基于模型的预测点和随机选取的数据点一样正确。

从图 4.20 中我们能估计出避障行为的李雅普诺夫指数：由于初始可用信息大约为 7.9bit，且预测时域为 80s，因此我们估计 $\lambda \approx \frac{7.9\text{bit}}{80\text{s}} = 0.1\text{bit/s}$。这与之前用沃尔夫法得到的结果相符（见表 4.2）。

这其中蕴含了很有趣的意义，例如，对于我们的机器人避障行为的计算机建模来说：不管一个模型做得有多好，都不可能预测大于 80s 以后机器人的精确轨迹（然而，该模型仍然可以对机器人与环境交互作用的动力学性质建模）。

2. 随机游走

图 4.21 给出了第三例的轨迹，即在稍复杂环境下的避障随机游走的轨迹。

对该例计算信息损失和预测时域的结果已经在图 4.22 给出。在大约 80s 以后，基线误差与模型误差之间相差就不大了，也就是说在这种情况下的预测时域为 80s。

在这种情况下，电脑所记录的 x 方向变化范围是 226。我们能够对机器人位置进行精确的点操作，这就意味着初始时能得到的关于机器人位置的信息量 $I_0 = \ln\frac{226}{2}/\ln 2 = 7.8\text{bit}$。因此，我们可以估计出 $\lambda_{rw} \approx \frac{7.2\text{bit}}{80\text{s}} = 0.1\text{bit/s}$。

信息损失曲线（图 4.22 底部）上并未显示出明显的线性区域，表明信息损失的指数模型与该部分数据吻合得并不好。

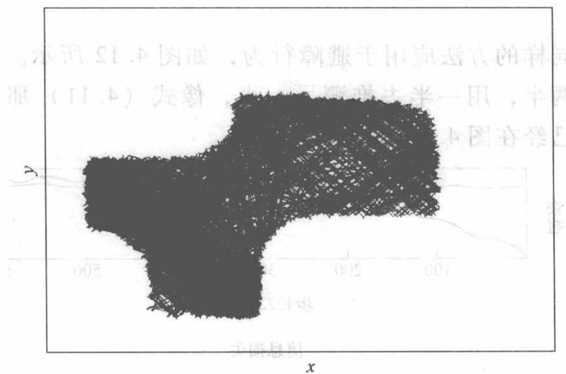


图 4.21 随机游走与避障

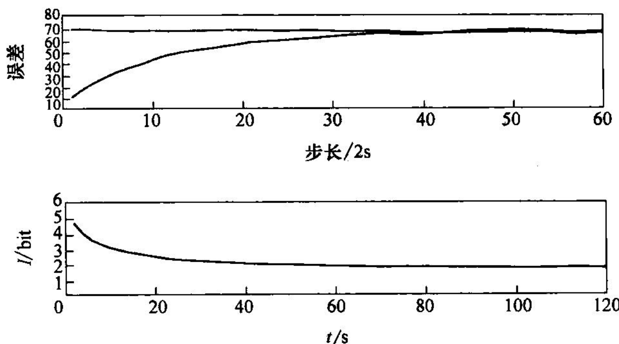


图 4.22 避障随机游走的预测时域和信息损失

4.4.5 用预测时域和吸引子大小估计李雅普诺夫指数

这里有一个可供选择的方法将李雅普诺夫指数与“真实的世界”联系起来，用的同样是机器人学实例中的墙壁沿行和避障行为。

之前是将李雅普诺夫指数 λ 定义为

$$D(t) = D_0 e^{\lambda t}$$

式中，$D(t)$ 是在时间 t 里在相空间中两点的间隔；D_0 是指当 $t=0$ 时在相空间中两点的一些微小的初始间隔。

当我们对机器人的位置只能用“在吸引子的某处”来描述时，可以说我们已经丢失掉了所有关于机器人位置的信息。显然，在 t_{crit} 这一点，初始误差 D_0 已经增大到整个吸引子 S 的大小，得到式（4.16）：

$$D_0 e^{\lambda t_{crit}} = S \tag{4.16}$$

我们可以用以下的简化假设，再利用图表例如图 4.19 和图 4.20 来估计李雅普诺夫指数。我们假设在相空间中所有的 N 点都是均匀分布的，且中间均间隔

D_0。第 n 点在 d 维吸引子上的每个维数如式（4.17）：

$$N = n^d \tag{4.17}$$

式中，d 是吸引子的维数。

从而由式（4.18）得出吸引子的边长 S：

$$S = D_0 n^d = ND_0 \tag{4.18}$$

从式（4.16）和式（4.18）得出式（4.19）。当我们的模型只能作出随机猜测而不是已知的预测时，式（4.19）能让我们用 t_{crit} 来估计李雅普诺夫指数：

$$\begin{aligned} D_0 e^{\lambda t_{crit}} &= ND_0 \\ \lambda t_{crit} &= \ln N \\ \lambda &= \frac{\ln N}{t_{crit}} \end{aligned} \tag{4.19}$$

将式（4.19）应用到墙壁沿行行为上，我们从图 4.19 可以看出我们数据驱动模型的预测误差基本不会变得平滑，这说明预测时域是一个无穷大值（实际上，预测时域会比 6000 大得多）。这意味着估计出的李雅普诺夫指数为 $\lambda \approx 0$，这和我们之前的计算相符。

在预测 600 步以后（$t_{crit} = 150s$），针对避障行为的数据模型驱动预测的误差曲线（见图 4.20）和基线误差会很接近。由于我们这里有 $N = 26000$ 个数据点，因此有 $\lambda_{oa} \approx \frac{\ln 26000}{150} s^{-1} = 0.07/s \approx 0.1/s$。这与我们之前用沃尔夫法估计出的结果相符（见表 4.2）。

对于这种方式的另一种说明是由我的同事凯斯·沃克（Keith Walker）依据 [Baker and Gollub，1996，p. 154f] 提出。

如果重构吸引子的嵌入相空间是 m，那么李雅普诺夫指数是和每一条主轴线相关联的。如果一个非常小的超球面点在一个 m 维空间内，并随时跟踪这些点的轨迹，初始的超球面点就会展开为一个由李雅普诺夫指数决定的形状。因为和主轴相关联的负李雅普诺夫指数会导致超球面的空间构成随着时间呈指数缩小，反之，正的指数则会导致扩展。超体积包含了时刻 t 的轨迹可表示为

$$V(t) = V_0 e^{(\lambda_1 + \lambda_2 + \cdots \lambda_m)t}$$

式中，V_0 是点在 $t = 0$ 时的超体积。

因此，对于一个有且仅有一个正李雅普诺夫指数的系统来说，起初的超球面体会展开成为一个椭球体。（注：如果存在两个正的李雅普诺夫指数，系统可参考超混沌系统。）

对于耗散系统，关系 $\sum \lambda_i < 0$ 保持不变，超体积随时间延续而趋近于零。如果其中一个李雅普诺夫指数为正（例如 λ_k）且其余的指数为负，得出的椭球体就会成为一条沿 k 轴方向的细纤维。这根纤维最终很可能会描述出这道轨迹。

如果超球面最初在 k 方向的维数是 d_k，那么这条纤维的延长线是 $d=d_k e^{\lambda_k t}$。既然吸引子是有限的，令 d_k 已经扩大为 D 时的时间为 t_{crit}。这就是说最初沿 k 轴依据 d_k 分开的两个点现在是依据 D 来分开。在这一时刻，系统的情形是从本质上不确定或是丢失的，关于这个系统，我们仅能知道它位于吸引子的某些地方，并且可知有 $\lambda_k=\frac{1}{t_{crit}}\ln\frac{D}{d_k}$。

[Baker and Gollub，1996] 中提到，已知 t_{crit} 并利用一个简化的假设，就可以做成一个合适的 λ_k 预测器。如果重构吸引子的系统中，一些参考量有 N 个大小，那么相近两点在相空间内的沿轴平均距离可以由 D/N 估计。如果令这一比值作为相空间内两点的初始不确定性 d_k，则

$$\lambda_k \approx \frac{1}{t_{crit}}\ln N \tag{4.20}$$

如果认为 $\ln N$ 随 N 缓慢地变化，则式（4.20）可以更进一步地简化为一条经验法则。例如：对于 $N=1000$，10 000，50 000，100 000 的不同取值，$\ln N$ 分别是 7，9，10.8，11.5。因此，对于绝大多数包含了 1000 乃至更多取值的数据，就可以概括地说

$$\lambda_k t_{crit} \approx 10 \tag{4.21}$$

将经验公式应用到墙壁沿行与避障例子中，我们得到了如下结果。对于有本质上无限大预期范围的墙壁沿行试验，t_{crit} 是无穷的，得出了一个估计结果 $\lambda_{wf} \approx 0$ [式（4.21）]。这和使用沃尔夫方法得出的数据吻合。

而对于避障行为，我们估计出 $t_{crit}\approx 150\text{s}$，得出一个估计结果：$\lambda=0.07\text{s}^{-1}=0.1\text{bit/s}$，这和之前得到的结论吻合。

4.5 非周期性：吸引子的维数

另一个动力学系统的主要特点显示确定性的混沌是指变化的状态不可精确回复到它们之前的状态。也就是说，系统的行为不具有周期性。相空间的轨迹被放在了一个具有不规则空间的吸引子（一个奇异吸引子 r）上。然而，从系统到另一个系统都是有变化的，不管状态变化回到前一个值多么接近，因此也就想要量化这个“接近”的程度。

对于不定期度的量化方法是吸引子的相关性维数 d。这个相关性维数显示了数据是否具有周期性和它的程度：周期性数据有一个为零的相关性维数，混沌的吸引子，有一个非整数的相关性维数 [Kaplan and Glass，1995，p. 321]。

下面介绍如何确定相关维数。

一个吸引子的维数和它的非周期性有关：力学上非周期性越强，吸引子维

数就越大。为了测量通过相空间的非周期性轨迹是怎样的，可使用下述方法。

假设在吸引子上任取一点，绕该点绘制半径为 r（即所谓的“相关距离”）的超球面，并数清有多少吸引子的点在这个超球面中。这些点的数目成为“相关积分” $C(r)$，由式（4.22）给出：

$$C(r) = \frac{\theta}{N(N-1)} \tag{4.22}$$

式中，θ 代表次数 $|D(t_i) - D(t_j)| < r$；I 和 j 是式（4.5）中嵌入的 D 中的两个不同的时间；r 是相关距离；$N(N-1)$ 显然是最大值。

当且仅当 $i=j$，$|D(t_i) - D(t_j)| < r$，理论上是可能的。

在一个完全周期性吸引子中，以理想摆为例，相关积分并不随 r 的增长而增长。$C(r)$ 对 r 的斜率是零。在其他情况中，$C(r)$ 随 r 的增长而增长。$C(r)$ 对 r 的斜率，即吸引子的相关维数。

在实际计算中，斜率通常依据式（4.23）进行估计［Kaplan and Glass, 1995, p. 354］：

$$d = \frac{\ln C(r_1) - \ln C(r_2)}{\ln r_1 - \ln r_2} \tag{4.23}$$

式中，r_1 粗略地取 $\sigma/4$（σ 是时间序列的标准差）；$C(r_1)/C(r_2) \approx 5$［Theiler and Lookman, 1993］。

显然，相关维数的计算依赖于选择的嵌入维数 p 和相关距离 r。在同一过程中，同时计算 p 和 d 的值是一个不确定的问题，目标是找到参数 p 和 r 的范围，d 几乎和 r 相等（一个所谓的“标度区间”）。换句话说，想找到一个用于计算相关维数 d 的区域，利用式（4.23），这并不精确地依赖于嵌入维数 p 和相关距离 r 的选择。

要找到一个标度区间，可以绘制相关维数 d 作为嵌入维数 p 在 1~10［Kaplanand Glass, 1995, p. 323］的相关距离 r 的函数，并检查是否有区域，使得其中 r 和 p 的选择不改变相关维数 d 的计算。这个 d 就是我们对吸引子维数的估计结果。

4.5.1 相关维数：机器人学举例

下面我们将估计图 4.12（右）中所述避障行为的相关维数。

首先，我们利用式（4.22）算出：$\frac{dC(r)}{dr} = d$，图 4.23 为显示的结果。

选择一个非常大的相关距离 r，比如像从很远的距离看一个物体：则会出现零维数［Kaplan and Glass, 1995, p. 323］。另一个极端是选择一个过小的 r，则会导致单一噪声的放大，两者同样不是我们想要的。

图4.23显示了一个标度区间——一个d不随参数变化的区域，存在大约$r=40$，在这个区域内，提高嵌入维数不再改变相关维数的计算。

要获得结论的另外一种表示法，我们可以使$r=40$不变，绘制d对p的图（见图4.24）。图4.24显示了嵌入维数在6～10时，相关维数d的计算结果变平了。相关维数d在这一点是$d\approx2.5$，一个分型维数意味着吸引子是奇异的，并且这一系统的行为是非周期性的，确定性混沌的特征之一。

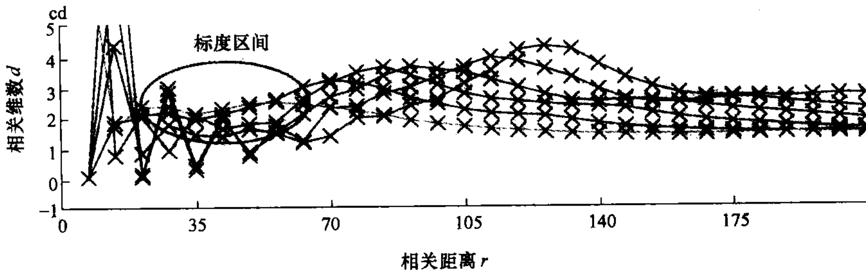


图4.23 对于图4.12中显示的避障行为的相关维数对相关距离的曲线（取嵌入维数5，7，9，11，13，15和多种相关距离）

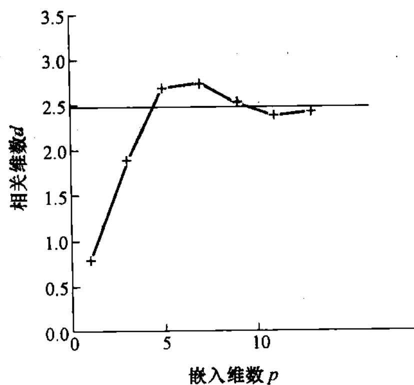


图4.24 对于图4.12（右）中显示的避障行为的相关维数对嵌入曲线（取$r=40$。相关维数计算结果是$d\approx2.5$）

图4.25显示了在墙壁沿行行为中相关维数d对p和r的曲线。我们估计得$d\approx1.4\sim1.6$。

随机游走

表4.3显示了当我们把式（4.23）应用到在4.4.4节介绍过的随机游走者

身上时得到的结果。

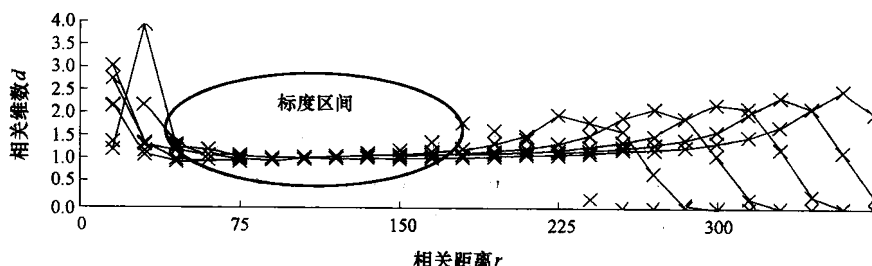


图 4.25　对于图 4.12（左）中显示的墙壁沿行行为的相关维数对相关距离的曲线（取嵌入维数 5，7，9，11，13，15，计算得相关维 $d \approx 1.4 \sim 1.6$）

表 4.3　使用不同嵌入维数 p 和嵌入相关距离 r 对随机游走中 d 进行估计

（计算应用式（4.23），嵌入延迟 $\tau = 40$）

p	r_1	r_2	$C(r_1)$	$C(r_2)$	d
3	15	12	0.0093	0.0058	2.12
		10		0.0039	2.14
		7		0.0018	2.15
4	15	8	0.0035	0.00072	2.52
		5		0.00022	2.54
5	20	10	0.0032	0.00048	2.74
6	15.8	20.7	0.0027	0.0066	3.31
8	15.8	17.3	0.00094	0.0012	2.69

根据表 4.3，我们估计随机游走的吸引子维数在 2.7 ~ 3.3。

4.6　小结

移动机器人的运动是时间的函数，是可被描述的，例如微分方程（如速度就是位移的导数）。所以，机器人实质上是一个动力学系统，本章内容采用了动力学理论来描述和分析机器人的行为。

动力学系统在物理空间中的运动可以完整地在相空间中表述。所谓相空间，就是一种用系统在每个自由度上的位置 $x(t)$ 与速度 $\dot{x}(t)$ 定义的一种空间。因为有了分析软件用于相空间的分析，这种方法很有吸引力。

用与所研究的动力学系统运行有关变量的观测值，通过时滞嵌入的方法（以时间为自变量，描述其它量与时间的关系），相空间可以重构。对于移动机器人，机器人的轨迹往往是与机器人行为联系最为紧密的参数之一，例如，用 x

轴，y 轴位置随时间变化关系，就可以重构机器人的相空间。

如果吸引子得到重构，它就可以量化地表述。可以通过以下三种定量的方法：

1）李雅普诺夫指数（4.4 节）。它测量 bit/s 单位内的信息损失，因此建立了这样的概念：微小的初始扰动放大得如此之快，以至于除了说“它在吸引子的某个地方”再没有别的语言来表述这个系统了。对移动机器人，这句话变成了“机器人将要到某个地方，但确定是哪里是不可能的”。

2）预测时域法（4.4.2 节）。这种测量方法与李雅普诺夫指数有关，基于模型它可以预测需要多少时间系统会变成未来的一个状态，一般情况下，精度就像随机猜测一样。

3）相关维数法（4.5 节）。这是系统周期性的测量：系统是否循环到以前的状态，还是相空间中的运动仅仅经过以前状态点的附近？

除了提供关于移动机器人与环境的交互作用的有效信息，诸如“还要多久可以实现用仿真的方法预测机器人的位置？”这样量化的描述在作为一门科学的移动机器人学中是必要的。其中另外一个原因是机器人行为从机器人、任务（控制程序）和环境之间的相互作用显现出来。如果可以量化地描述机器人行为，如李雅普诺夫指数、预测时域法、相关维数法，对研究者来讲，一种新的机器人实验方法就变得可行了。保持三个元素中的两个不变，用量化的测量方法系统的研究第三个元素对机器人行为的影响。机器人学研究应该包括量化的东西，而不是定性的估计，允许独立重复以及对实验结果的验证。这是成熟科学的标志。

第 5 章　智能体行为的分析——案例研究

概要：本章将第 4 章中出现的方法应用到真实世界实验数据中。我们将分析三个数据集：在周围空间中随机移动的机器人的运动、混沌步行者和信鸽的飞行路径。

5.1　随机移动机器人的运动分析

图 5.1 是不规则环境的一个麦哲伦专业版移动机器人 3 个小时的运动轨迹。

图 5.2 中用 $x(t)$ 和 $y(t)$ 分别表示顶部和底部的运动轨迹。

我们对分析这个机器人的行为非常感兴趣：它是确定性的还是随机的呢？它可预测么？如果可以，那么它还会向前走多少步？它的这种行为怎样才能量化地描述呢？

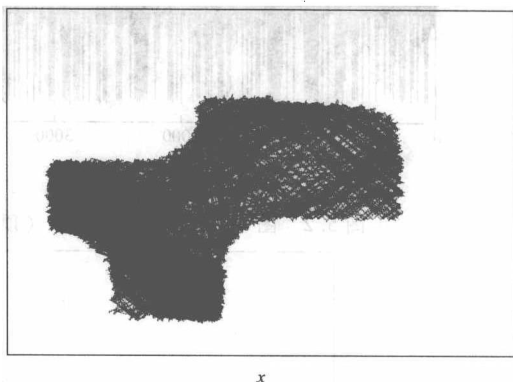


图 5.1　不规则环境的一个麦哲伦专业版移动机器人 3 个小时的运动轨迹

5.1.1　确定性

为确定 $x(t)$（见图5.2）是否主要是确定的，我们采用 4.3.2 节中介绍的方法。

$x(t)$ 和 $x(t-3)$ 的回归图（见图 5.3）表明 $x(t)$ 是确定性的而不是随机的（将图 5.3 与图 4.11 进行对比）。

用三维嵌入的方法将 $x(t)$ 的第一半作为第二半的预测，确认了这个结果，如果用信号的均值来预测，将预测误差 ε_{model} 和基线预测误差 ε_b 比较。比值 $\varepsilon_{model}/\varepsilon_{mean}$ 结果是 0.048，比 1.0 小得多，由此 $x(t)$ 确实是确定的。

5.1.2　稳定性

接下来，我们将确定 $x(t)$ 是否是随机的，用 4.3.2 节中介绍的游程检验方法。任意的将 $x(t)$ 分成 110 个区域，我们来决定是否每个区域里的中值都大于或小于整个 $x(t)$。

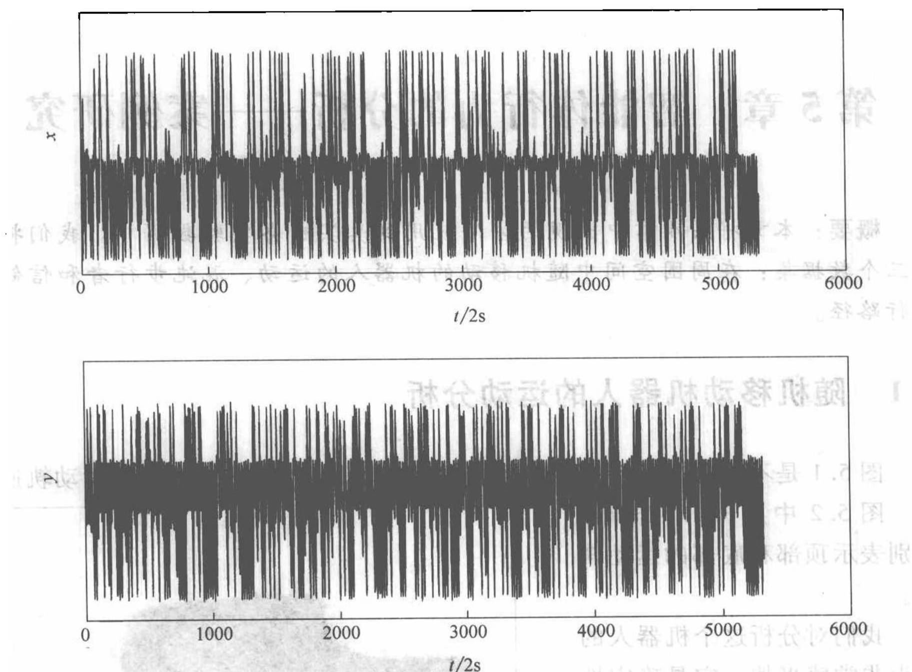


图 5.2　图 5.1 中的轨迹分为 x（顶部）和 y（底部）轨迹

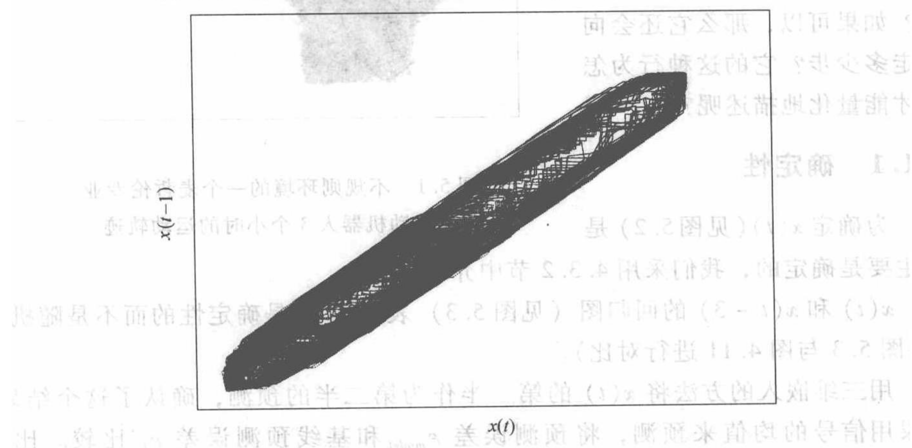


图 5.3　随机游走机器人 $x(t)$ 的回归图

结果发现有 55 条轨迹，这个结果恰好在可以接受的范围［45，65］之内，表明高于中间值和低于中间值的分布是随机的：$x(t)$ 是稳定的。

"$x(t)$ 是稳定的"这个结果是由变化的非参数分析确定的。对于 x（1～3400s）、x（3400～6800s）和 x(6800～10400s)($p=30$)，它们之间没有显著差异。

5.1.3 随机游走避障的可预测性和李雅普诺夫指数

根据第4章的介绍可以确定，我们的数据满足所讨论的方法，现在我们要试着对数据进行量化的描述。正像在4.4.2节介绍的，我们先看一下 $x(t)$ 的可预测性。

为了确定预测时域 $x(t)$，我们需要确定准确的重构吸引子的嵌入延迟。当自相关达到 1/e [Kaplan and Glass, 1996]，或互信息到达它们第一个极小值 [Abarbanel, 1996] 时，上述嵌入延迟就可以得到确定。$x(t)$ 的自相关和互信息如图5.4所示。

自相关降到 e^{-1}，此时 $\tau=9$，互信息到达其第一个极小值，此时 $\tau=7$。我们选择8作为嵌入延迟。

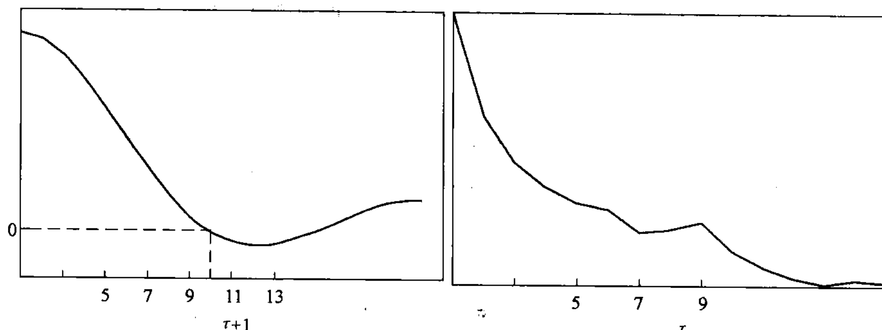


图5.4 随机游走机器人的自相关（左）与互信息（右）

现在我们来预测 $x(t)$ 的另一半。把第一半作为模型，用每一个 $x(t)$ 点来构建一个三维空间嵌入 $D=[x(t),x(t-8),x(t-16)]$，然后找到我们前一半数据里每个 D 最近的邻近点 D_n。D_n 的后继点可以用来预测 $x(t)$ 的后继点。

图5.5显示的是结果。我们可以知道，因为可以预测到35步以前，所以使用我们数据的平均预测误差，和使用前一半数据的一个随机选取点来预测后一半数据的误差是一样大的。换句话说：如果你要预测随机游走移动机器人的精确位置（其轨迹如图5.1所示），那么平均起来看，这样比随机预测得好，预测时域是70s。

图5.5同时给出 $x(t)$ 的李雅普诺夫指数的估计。正如我们前面确定的，预测时域是70s。因为我们最初有5.2bit信息（图5.5的下面），我们估计李雅普诺夫指数为 $\lambda\approx\frac{5.2}{70}=0.07$ (bit/s)。

通过使用沃尔夫的方法 [Wolf, 2003]，我们可以获得统计估计 $0.07\text{bit/s}<\lambda<0.1\text{bit/s}$，这证实了我们的结果。

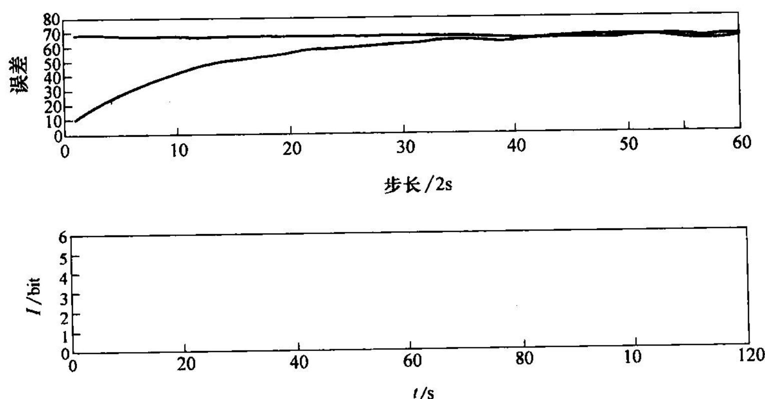


图 5.5　随机游走机器人绕过障碍物的 $x(t)$ 的预测时域和信息丢失

5.1.4　吸引子的分析

已经确定了随机游走行为表现出确定性的混沌，我们将分析机器人的相空间。首先，我们使用 4.2.3 节介绍的时滞嵌入方法来重构吸引子，结果如图 5.6 所示。

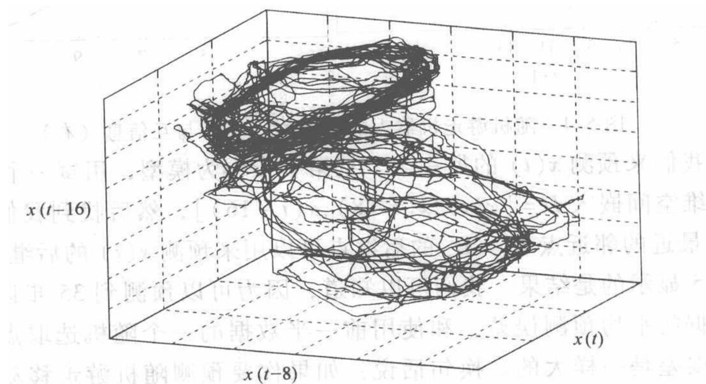


图 5.6　随机游走行为的相空间的三维重构表明了一个吸引子，它由两个闭环组成，而且其中一个要比另一个定义得更清楚

现在通过估计图 5.6 所示吸引子的维数，可以确定机器人的运动是否是周期性的。

现在已经确定机器人的随机游走避开障碍物行为是确定性的（这没什么好惊奇的，因为机器人的当前位置是由机器人之前的位置决定的，机器人物理上不能随机到处跳跃）和稳定的。我们已经估计出预测时域大约是 70s，同时李雅普诺夫指数大约是 0.07bit/s。这些是对随机游走机器人避开障碍物行为的吸引

子的两个定量描述，而且我们可以使用第三个，即相关维数。

随机游走机器人相关维数的计算已经在 4.5.1 节中介绍过了。然后我们估计出 $2.7 < d < 3.3$，表明随机游走行为是非周期性的。

5.1.5　随机游走：小结

总的来说，我们发现随机游走机器人的行为（见图 5.1）是确定和稳定的。我们先前介绍过，如果我们有机器人对环境反应的定量描述会有好处。现在我们有三个：这种行为的预测时域大约是 70s、李雅普诺夫指数大约是 0.07bits/s、相关维数是 3.0。

5.2　混沌步行者（Chaos Walker）

图 5.7 表明麦哲伦机器人 Radix 的运动轨迹是在编程中使用了图 5.1 中的策略。本质上，这种策略包括一个短时间直线运动，然后是在端点上转弯，其转弯方向和转弯持续时间是由式（4.14）中给出的二次迭代程序决定的。

机器人的位置每隔 250ms 被记录一次。机器人所在的环境如图 5.8 所示。

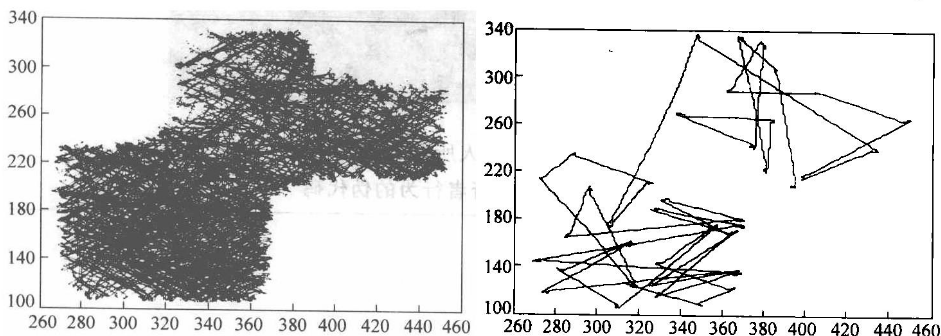


图 5.7　“混沌步行者（Chaos Walker）”的轨迹
（完整的轨迹显示在左图，5min 的轨迹显示在右图）

机器人 Radix 预先沿一条直线微微地移动了 3s，然后在端点上转弯，其时间由式（4.14）给出的二次迭代方程中的混沌方程确定。我们要分析这个行为，然后定量地描述它。为了实现这个目的，我们将使用机器人沿 x 轴的运动，这个运动如图 5.9 所示。

5.2.1　稳定性

首先，我们确定 $x(t)$ 是否是稳定的。把整个时间数列 $x(t)$ 分成三个相同大小的区域，我们通过分析，三个部分的均值和标准差的非参量是不同的（见

3.4.4 节），意味着信号是不稳定的。

表 5.1 为混沌步行者行为的伪代码。

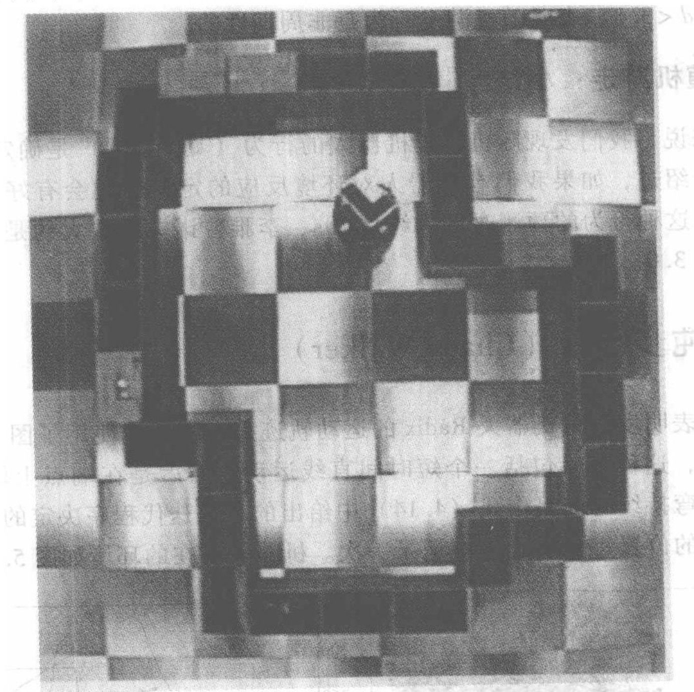


图 5.8 机器人所在的环境

表 5.1 混沌步行者行为的伪代码

```
d(1) =0.125
t=1
while(1)
        t=t+1
        If obstacle detected
                Perform standard obstacle avoidance action
        else
                Move forward for 3.3 s
                d(t) =4d(t-1)(1-d(t-1))
                if d(t) >0.5
                        TurnTime = (PI * d(t)/0.15) seconds
                else
                        TurnTime = -(PI * d(t)/0.15) seconds
                Turn on spot at constant speed for TurnTime s
        end
end
```

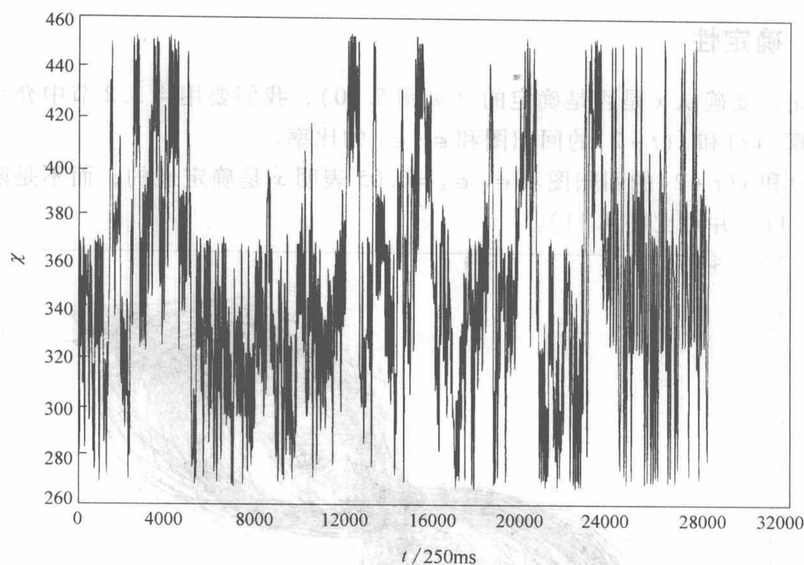


图 5.9　机器人沿 x 轴的运动

然而，正如前面所讨论的，信号$\frac{\mathrm{d}x}{\mathrm{d}t}=\dot{x}$通常是稳定的。这个信号如图 5.10 所示，并且，用变量的非参数分析法可以证明确实 $\dot{x}$ 是稳定的。因此我们将研究 $\dot{x}$。

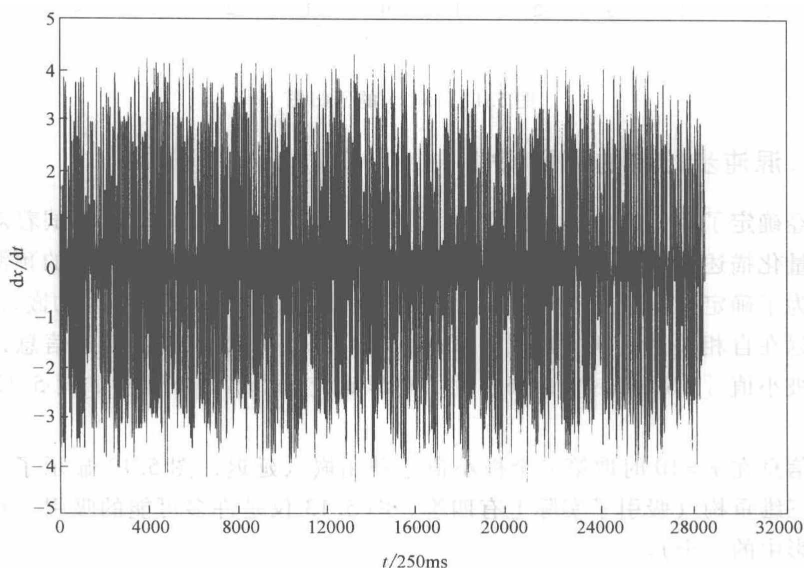


图 5.10　5.7 图所示运动的 x 的一阶导数

5.2.2 确定性

首先，要确认 $\dot{x}$ 是否是确定的（见图 5.10），我们要用 4.3.2 节中介绍的方法来计算 $\dot{x}(t)$ 和 $\dot{x}(t+2)$ 的回归图和 $\in/\in_b$ 的比率。。

$\dot{x}(t)$ 和 $\dot{x}(t+2)$ 的回归图和 $\in/\in_b=0.05$ 表明 $\dot{x}$ 是确定性的，而不是随机的（见图 5.11，并对比图 4.11）。

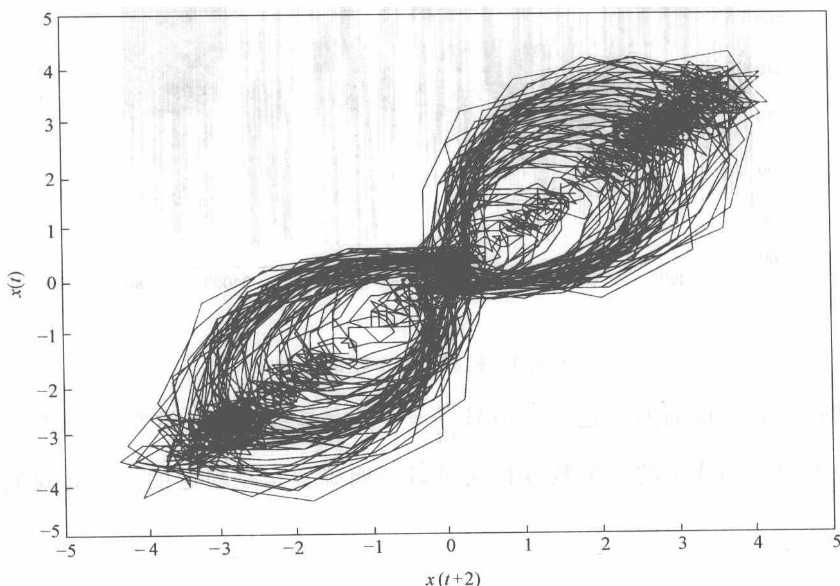


图 5.11　$\dot{x}(t)$ 的回归图

5.2.3 混沌步行者的可预测性

已经确定了我们的数据服从第 4 章中讨论的规则，我们现在将试着对数据做一些量化描述。正如 4.4.2 节中介绍的那样，我们将通过对 $\dot{x}(t)$ 的预测开始实验。为了确定 $\dot{x}(t)$ 的预测时域，我们需要确定重构吸引子的正确的嵌入延迟，这个延迟在自相关达到 1/e 时 [Kaplan and Glass, 1995] 或者当互信息达到它第一个极小值 [Abarbanel, 1996] 时，可以确定。$\dot{x}(t)$ 的互信息如图 5.12 中所示。

互信息在 $\tau=10$ 时取第一个极小值。利用嵌入延迟，图 5.13 显示了一个吸引子的三维重构（吸引子实际上有四维，图 5.13 仅是许多可能的吸引子在三维空间投影中的一个）。

现在我们用前半部分作为模型，通过构造每个点 $\dot{x}(t)$ 的三维坐标 $D=[\dot{x}(t),\dot{x}(t-10),\dot{x}(t-20)]$，并且在前半部分的数据中找出距离每个 D 最近

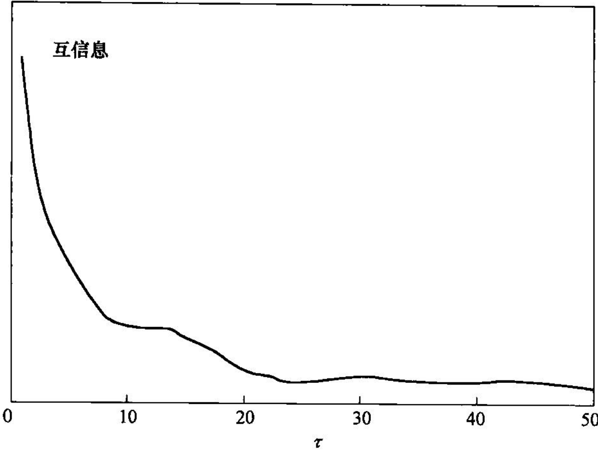


图 5.12 混沌步行者的 $\dot{x}(t)$ 的互信息

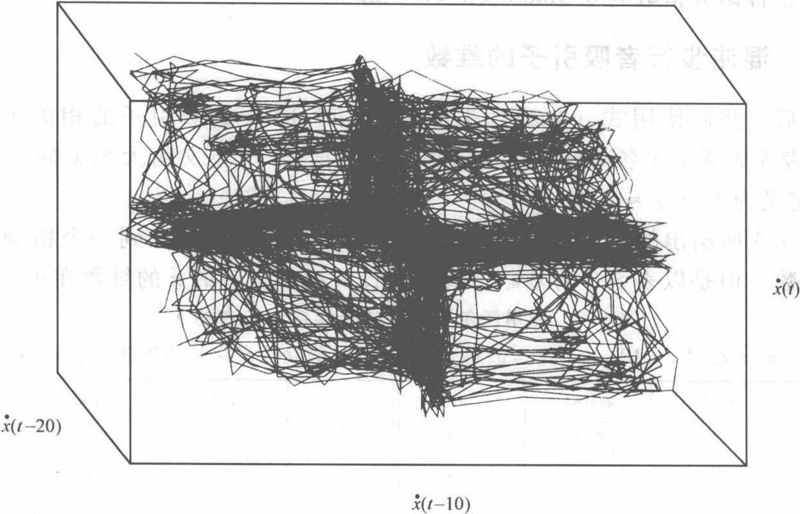


图 5.13 混沌步行者吸引子的重构（其运动上沿着 $\dot{x}$ 轴，$\tau=10$）

的 D_n，以此来预测 $\dot{x}(t)$ 的后半部分。D_n 后面的点用来预测 $\dot{x}(t)$ 后面的点。

结果在图 5.14 中表现出来。预测的时域位置在 19s 左右，而且最初约有六个具有可用信息的点，我们通过预测的时域位置来估计李雅普诺夫指数 $\lambda=6/19\approx0.3$（bit/s）。

数据信息的减少表现为一个线形区域（在图 5.14 中可见），区域内 λ 信息的减少大约为 $\lambda=4.3/8\approx0.5$bit/s。

为了得到一个推断，我们使用沃尔夫算法来判断 λ，结果在表 5.2 和表 5.3 中列出。

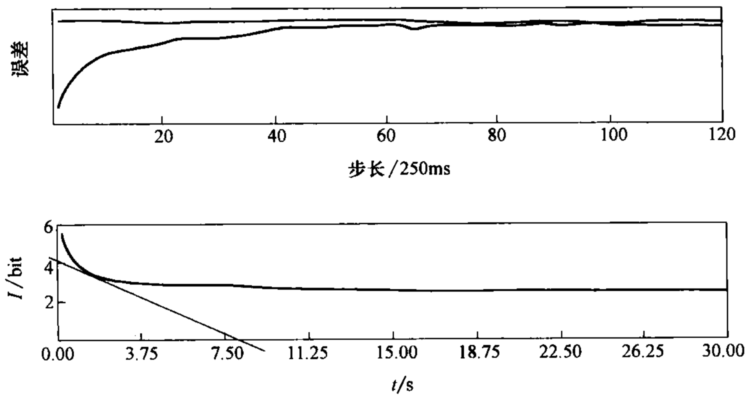


图 5.14 $\dot{x}$ 的预测时域（预测时域在 19s 左右）

使用沃尔夫算法，我们推断出 $\lambda \approx 0.4$bit/s，因此我们最终推断的混沌步行者的李雅普诺夫指数为 0.3bit/s < λ < 0.5bit/s。

5.2.4 混沌步行者吸引子的维数

最后，我们使用式（4.23）来推断在图 5.13 中的吸引子的相关维数。表 5.2 和表 5.3 展示了各种嵌入维数 p 和相关距离 r。r_1 定义为大约 $\dot{x}$ 标准差的 1/4，r_2 定义为 $C(r_1) \approx 5C(r_2)$。

表 5.3 所给出的结果表示了不显著的标度区间，很难找到一个精确的吸引子的维数，但是以表 5.3 为基础，我们估计混沌步行者的 $\dot{x}$ 的维数在 4 左右。

表 5.2 使用沃尔夫算法估计混沌步行者的 $\lambda_{\dot{x}}$

（嵌入延迟 $\tau = 10$，scalmn = 0.1。预测时域内的参数设定是用黑体字表示的）

p	*evolv*	scalA1	λ	p	*evolv*	scalA1	λ
3	3	0.8	1.9			0.9	0.7
	5		1.4			1.1	0.6
	7		1.2			1.3	0.6
	9		0.8			1.5	0.6
	11		0.7			1.7	0.6
	13		0.7			1.8	0.6
	15		0.7			2.2	0.5
	17		0.6	3	13	1.4	0.6
	19		0.5	4			0.5
3	13	0.3	1.0	5			0.4
		0.4	0.9	6			0.4
		0.5	0.8	7			0.4
		0.7	0.7	8			0.4

表 5.3　对应的 $\dot{x}$ 在距离 r 上的积分以及混沌步行者的 $\dot{x}$ 的相应维数(式 4.23)

p	r_1	r_2	$C(r_1)$	$C(r_2)$	d
3	1.77	0.8	0.101	0.023	1.9
4	1.8	0.9	0.0443	0.0079	2.5
5	1.8	1.0	0.0179	0.00297	3.1
5	2.3	1.4	0.038	0.008	3.1
6	1.8	1.2	0.00695	0.00167	3.5
6	2.5	1.8	0.024	0.00695	3.8
7	1.8	1.2	0.0027	0.000587	3.8
7	2.5	1.8	0.011	0.0027	4.3
7	3.5	2.5	0.05	0.011	4.5
8	1.8	1.2	0.0011	0.00028	3.4
9	1.8	0.6	0.0005	0.00014	1.2

5.3 信鸽飞行路径的分析

驯化的岩鸽具有一种惊人的能力，当它们被放飞在完全陌生的、距离鸽巢数百公里之外的地方时，它们依然能够返回鸽巢。它们运用先天就有的磁性指南针和后天学习的太阳指南针、导航知识（例如地磁场的磁力分布）来实现这一目标［详见 Wiltschko and Wiltschko，2003］。尽管大多数鸽子能够展示出可靠的归巢能力，但是在不同的单次之间存在差异。很多因素对于鸽子的归巢能力都是有影响的，例如它们是否认识被放飞的地点；某只鸽子是否具有经验、天气如何、沿途是否遇到其他鸽子等。以上因素中的一部分是源于环境（分析鸽子行为时，这些因素是不容易被考虑的），但是其他因素是对于不同单次而言是详细而具体的，并且，运用本书中的方法强调不同单次之间的行为区别将是非常有趣的。

在这个案例研究中，我们将比较两只鸽子的归巢行为。我们的目标是鉴别这两种归巢行为的相同点和不同点，也就是描述鸽子的行为特征。

5.3.1 实验步骤

为了进行研究，我们将一个微型的 GPS 定位装置固定在鸽子身上。接着把鸽子带到距离鸽巢大约 21km 的放飞地点，然后放飞。它们在飞回鸽巢的过程中，每一时刻的位置（经度和纬度）都被记录下来。然后将这些坐标值转换成固接于地球的地心坐标（ECEF）。我们要分析的这两个飞行轨迹如图 5.15[⊖]所示。

⊖ 我非常感谢德国法兰克福大学（J. W. Goethe University of Frankfurt）的凯伦·冯·胡奈尔宾（Karen von Hünerbein）、罗斯维塔（Roswitha）和沃尔夫冈·威尔兹克（Wolfgang Wiltschko）提供这些可用的数据。

569 号信鸽和 97 号信鸽在离鸽巢相同距离的地点被放飞，并且都经过一条相当笔直的路径成功地返回鸽巢。因此，它们彼此似乎非常相似，但是，下面的分析将会显示它们行为中重要的不同之处。

5.3.2 分析

在分析图 5.15 所示的飞行路径之前，先声明：我们分析的路径是这样的，获取数据时不考虑当时的环境条件。我们将假设整个的飞行路径包含有意义的信息——信鸽单次的飞行特征。我们将证明，这两种飞行路径显示了两种具有不同特征的截然不同的状态。为便于分析，我们在这里假设这两种不同的状态都是信鸽行为的相关部分，而不是因为一些其他实验参量，例如天气、光线、噪声或者其他此类因素。

为了分析图 5.15 所示的路径，需要提出的第一个问题是："哪一个数据包

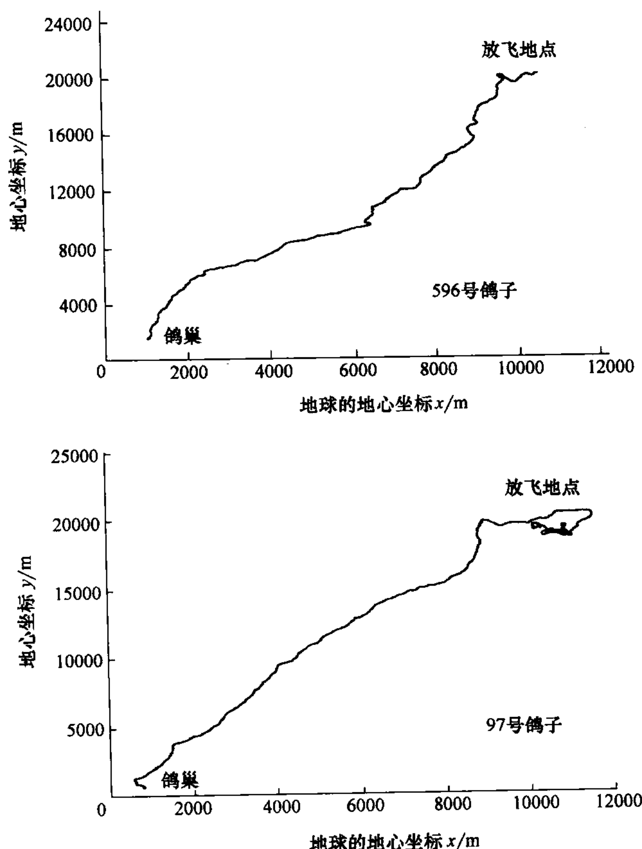


图 5.15　569 号信鸽（上）和 97 号信鸽（下）的归巢飞行路径

含了信鸽行为的相关信息?” 当我们观察这里的归巢行为时，一个有意义的数据是任意时刻信鸽与归巢方向的偏差，即信鸽飞行方向与归巢方向的差异。这些差异如图5.16所示，图中的零度虚线表示归巢方向。[⊖]

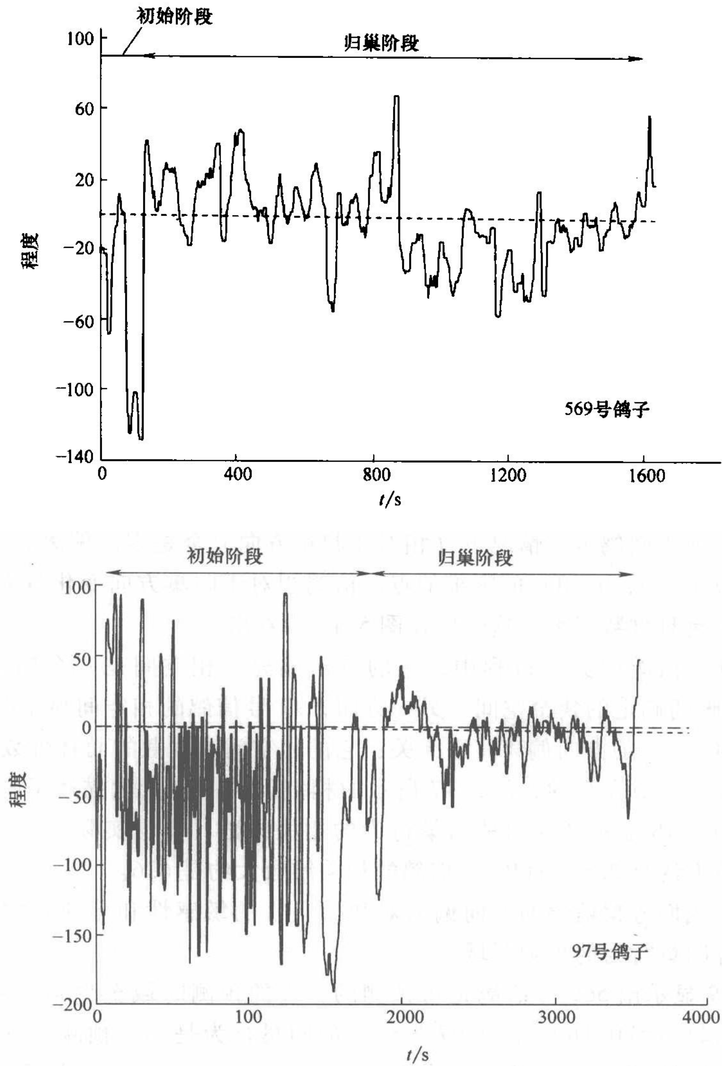


图5.16 信鸽与归巢方向的偏差（注意：时间刻度是不同的）

为了将第4章讨论的分析方法应用于图5.16所示的时间数据序列，我们首先检验它们的确定性和稳定性（参见4.3.2节）。结果是，两组偏差数据都是确

⊖ 图5.16所示数据是经过19s时间窗的中值滤波后的数据。

定性的（569 号信鸽 $\in/\in_b=0.29$，97 号信鸽 $\in/\in_b=0.23$），但是只有 569 号信鸽的偏差数据能够通过关于稳定的检验（参见 4.3.2 节）。97 号信鸽的偏差数据对于两个单独的区域分别是稳定的（初始化状态和归巢状态），但是对于整个区域不是稳定的。然而，它是弱稳定的，因为整个区域的均值是粗略相等的。由此，我们可以得出如下结论，两只信鸽在行为上不同，对于 97 号信鸽的结果必须谨慎地说明。

图 5.16 中，569 号信鸽和 97 号信鸽之间一个显著的区别显示出来：开始的时候，这两只信鸽都在放飞地点花费了一定时间，569 号信鸽花费不到 200s 之后，径直飞回鸽巢（即它的飞行方向相对于零度线或多或少地对称变化），然而 97 号信鸽花费了 10 倍的时间之后，才飞向鸽巢。我们并不知道其中的原因——这可能是因为 569 号信鸽更加善于飞行，同样可能因为像 97 号信鸽这一类信鸽就是在放飞地点需要花费较长时间之后，才能飞回鸽巢。无论原因如何，信鸽在放飞地点的不同行为是显而易见的。

如果我们对于信鸽的归巢行为进行动力学分析，这种区别同样会显示出来。关于这种区别最明显的图解是图 5.16 所示数据的相空间的重构，如图 5.17 所示。通过时滞嵌入重构相空间揭示了在一段时间上偏离之间的关系：在一个信鸽完全径直地飞回鸽巢的情况下（相对于归巢方向完全是零度偏离），它的相空间是一个位于（0，0，0）的零维的点。信鸽相对于归巢方向变化越大，则相空间越模糊失真且维数越大。这可以在图 5.17 中看出。

在 569 号信鸽归巢的过程中，它的方向偏离的相空间是一个围绕（0，0，0）点的清晰的确定的狭窄空间。另一方面，97 号信鸽的相空间显示时间 t 的方向偏离与时间 $t-\tau$ 的方向偏离几乎无关：它的相空间是发散的而且维数大的。

如前面 4.5 节介绍的，图 5.17 所示的相空间的一个量化描述子是它们的相关维数。图 5.18 显示了这两种归巢行为的相关维数计算，实际上，569 号信鸽的相关维数大约是 2.3，而 97 号信鸽的相关维数大约是 6.6。

最后，我们考察信鸽的方向偏离对初始条件的敏感性和可预测性。图 5.19 显示了两种归巢行为的预测时域。

图 5.19 显示出 569 号信鸽是可预测的，它的预测时域至少是 40s，它的李雅普诺夫指数不到 0.1bit/s，而 97 号信鸽的归巢行为是不可预测的（它的预测时域大约至少是 15s，它的李雅普诺夫指数大约是 0.3bit/s）。如果运用沃尔夫算法 [Wolf，2003]，我们得出对于 569 号信鸽 $\lambda\approx0.1$bit/s，对于 97 号信鸽 $\lambda\approx0.3$bit/s。

5.3.3 概要和结论

在上一个案例的学习中，我们比较了两只信鸽的自导航行为。更确切地说，

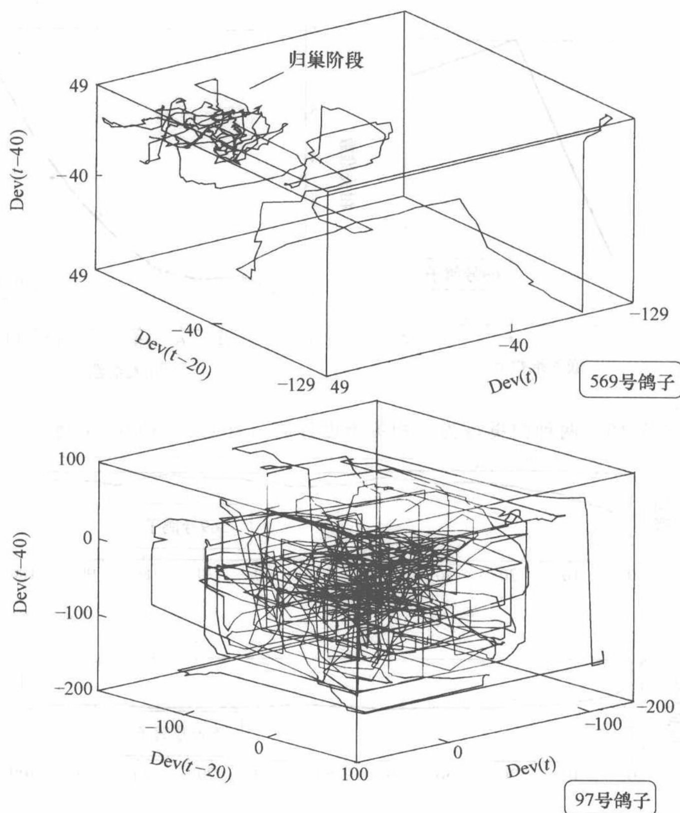


图 5.17　图 5.16 所示数据的相空间的重构

我们比较了两只鸽子飞行中对主方向的偏离随时间的变化情况。

虽然两只鸽子都成功地返回了鸽巢，同时它们的飞行路径（见图 5.15）看起来也非常相似，但是很快我们就发现它们归航的行为有着相当大的差别。第569号信鸽在离开放飞处不到200s后，就能径直地飞向目的地，而第97号信鸽则花了超过了1400s才做到这一点。正是这个初始化时间段区分了两只信鸽的不同，而要单看它们自导引飞行的阶段它们确实很相似。

假设初始化阶段是鸽子行为的一个描述性部分，而不是试验的偶然事件，我们重构和分析两只鸽子偏离实际回家路径的相空间，不出意外地可以发现吸引子描述出了两只鸽子归航引行为的不同之处。

第569号鸽子的吸引子的相关维数在2.3左右，显示鸽子归航行为具有相应的清晰定义的区间。而第97号鸽子的吸引子是发散的，它的吸引子的相关维数为6.6左右，没有明显的归航的区域。

除此之外，第569号鸽子的归航行为能够很好地预测（$\lambda<0.1$bit/s），而第

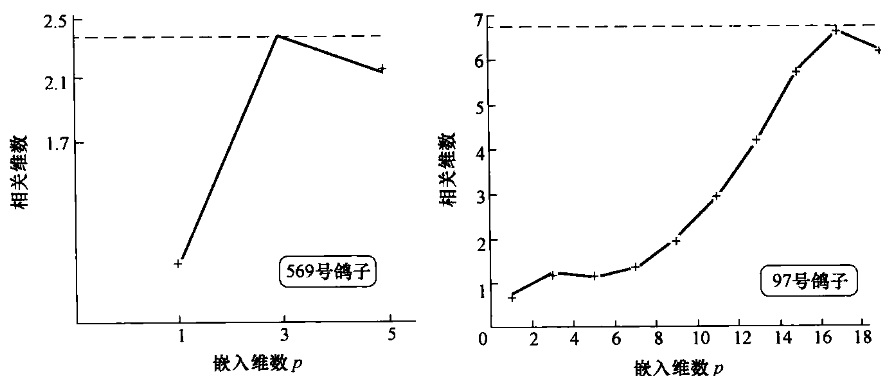


图 5.18　两种归巢行为的相关维度计算（注意：不同的比例尺）

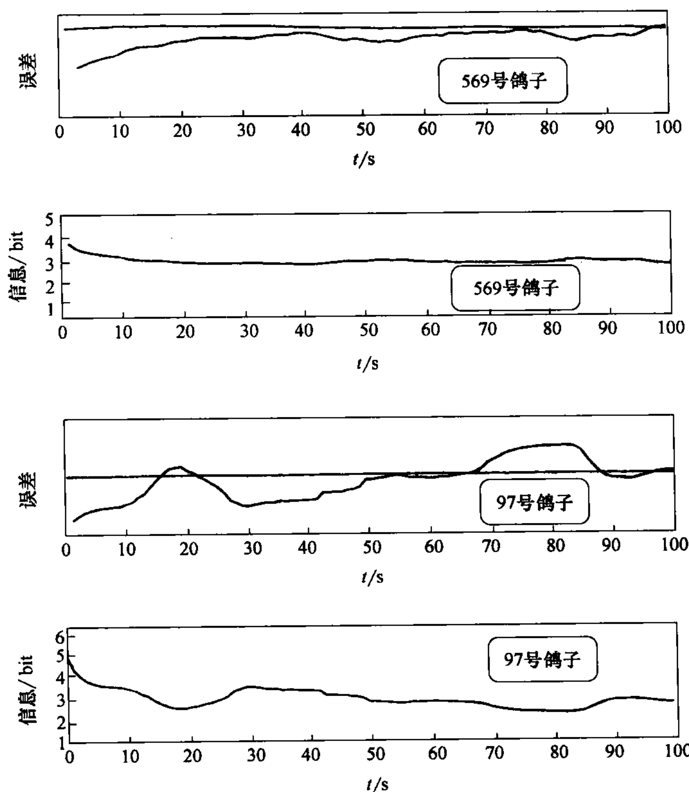


图 5.19　两种归巢行为的预测时域

97 号鸽子的归航行为是高度不可预测的（$\lambda = 0.3$），并显示出确定性的混沌的特征。

第 6 章　机器人—环境交互的计算机建模

概要：这一章介绍了一般的机器人—环境交互建模（比如动机、数据记录、采样、时间序列分析），并利用系统辨识技术，比如人工神经网络和 ARMAX（具有外部输入的自回归滑动平均模型）、NARMAX 模型（非线性 ARMAX），对机器人和环境交互进行建模。

6.1　引言

在这一章中，将研究三种不同的机器人与环境交互作用的建模方法，获得机器人运动的相关参数的数学描述。开展这项工作主要有两个意义。一是建模能为我们提供机器人基本特征的计算机仿真，这简化了机器人控制代码的开发；二是模型能够抽象地保持那些机器人运行时重要的方面，因此为我们分析行为和科学机器人学提供了一个基本的工具。

1. 动机——可靠性仿真

进行移动机器人实验是一项非常费时、昂贵和困难的事情。由于机器人—环境交互作用的复杂性，不得不重复做实验，以得到统计上有意义的结果。同时，由于是由机械装置和电子元件构成的机器，机器人并不能够在每次实验中都表现得一样。随着参数的变化，机器人的行为有时也会产生戏剧性的改变。比方说由于光滑表面的镜面反射作用，或是环境参数的影响比如灰尘、电池电量改变，导致电动机特性的改变等。

这些和硬件有关的问题使仿真成为一个非常有吸引力的选择。如果能够在数学模型中抓住那些决定着机器人—环境交互作用的根本因素，就能够用计算机而不是机器人来预测那些实验的结果。这种方法更加快速、更加便宜，同时还有另外的好处，那就是能够用精确的参数不断重复实验。这样就能够使使用者研究单一变量对机器人行为的影响，这在真实机器人身上是做不到的（因为真实世界中没有两种完全一样的情况）。

除了精确的可重复性，迅速、简便和低成本，仿真还有很多其它的优点。由于能够提供一个可靠的模型，仿真成为一个用来预测那些太过复杂的、不易于进行理论分析的行为，或是至少目前缺乏足够数据来进行严格实验的系统的行为（比如在第一个人进入太空之前对太空的探索）。仿真允许我们控制参量的变化，正是这种改变又能回过头来，帮助我们更好地理解这个模型。仿真可以

用来教学、学习和激发兴趣（比如说游戏）。我们能够利用模型来分析假设的情形，并且仿真就能够帮助我们加深对“如何把一个复杂系统分解成为若干子元素”的理解。

2. 作为科学工具的模型

在机器人学中一个重要的科学方法就是去理解机器人—环境的交互作用，找到支配机器人行为的主要的因素，同时为机器人的运行做出预测。

机器人—环境交互作用是高度复杂的、经常是混沌的过程，以致研究很难在真实世界中的真实的机器人上进行。因此机器人建模的动机之一就是建立一个抽象的、简单的机器人—环境交互作用的模型用于进行严格的分析。

这不是一种新方法。斯康隆（Schner）和凯尔索（Kelso）在1988年证明了利用随机非线性动力学的概念来理解行为模式，特别是生物行为模式的可能性。在很多情况下，复杂的行为能够用维数较低的动力学来描述，结果得到更少的自由度，从而实现了系统分析的简化（从属原则）。机器人建模的另外一个动机是得到一个简单的、易懂的、易于分析的机器人—环境交互作用的模型。

3. 基本的建模方案

基本的建模方案如图6.1所示。输入向量 $u(t)$ 与输出向量 $y(t)$ 相关；建模过程的目标就是去“辨识” $u(t)$ 和 $y(t)$ 的递归关系。因此“系统辨识”这个术语就用于这一过程。

如果仿真的目标是为了获得一种建模系统的可靠表示，那么由于真实世界的不可预测性和机器人对于噪声和变化的敏感性，建模需要真实数据来构造，而不是关于机器人和环境的一般假设。当然，这就意味着，每一个模型只能对在一个特定环境下运行的一个特定的机器人进行建模（见图2.1）！

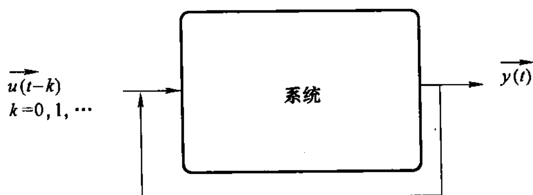


图6.1 基本的建模情形

有很多种可以实现这一建模任务的可能性。其中一种捷径就是采集真实世界中不同位置的数据，在预测阶段，用插值的方法来预测感兴趣的数据。插值的难点在于必须预先存储大量的数据，它们和已知点的间隔越近，插值就越精确。很明显，能被采集和存储的数据的量是有限的，所以总是存在准确和效率之间的折中。

（至少）存在第三种对 $u(t)$ 和 $y(t)$ 之间关系建模的可能性：使用人工神经网络（在6.3节会介绍）、线性多项式（6.4.2节）和非线形多项式（6.5节）。对于数据的记录和采样率的讨论之后，我们将依次讨论这些可选方法。

6.2 关于机器人建模的一些实际考虑

下面介绍一个确定采样频率的数据采集实例。

在这个例子中我们来确定一个合适的采样频率，用于分析墙壁沿行机器人的行为。机器人的轨迹如图6.2所示。我们想要分析的是$x(t)$（图6.2的右图）。

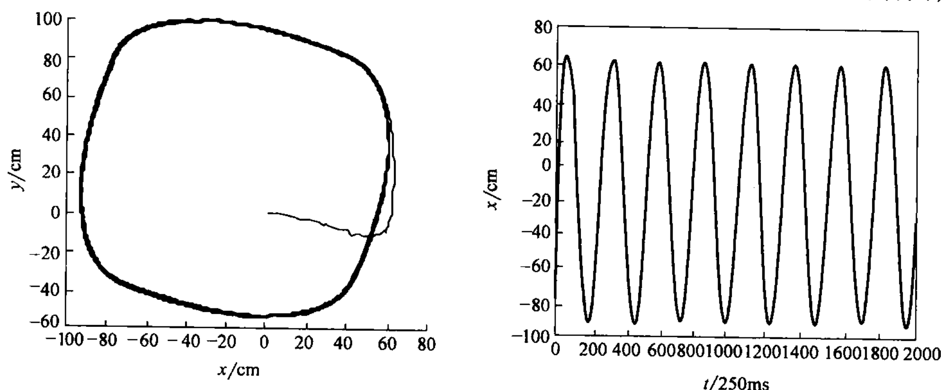


图6.2 墙壁沿行机器人的运动，由一个顶置的摄像机记录的情况（左图）和运动中x坐标随时间变化曲线（右图）

在这个例子中，数据的采样频率为每250ms一个。问题是：这个采样频率在这个实验情形中是否充分，或者是不是太高或太低了？如果是正确的，那么明显的$x(t)$可以直接分析出来。如果太高（超采样），可以简单地通过乘以一个系数$1/n$（只用于n倍的数据点时）减小样本数据。如果采样频率低，只能重复实验，然后用一个高一些的采样频率来记录机器人的行为。

下面是四种确定采样率是否合适的方式：

1）分析自相关性。

2）分析互信息（见式4.6）。

3）分析频谱。

4）使用一般的物理分析。

明显地，当采集数据时，关键是只采集“有意义”的数据点，即一个新的数据点应该包含新的、相关的信息。如果采样频率太高，后续的数据点实质上包含着相同的信息，因而是无意义的。目标是发现后续数据点包含新信息时的采样频率，可是还要获得数据的基本特征。

上述所列的前两点可以通过计算那些$x(t)$和$x(t+\tau)$相关性接近于0的数据点来做到这一点。图6.3同时显示了$x(t)$自相关性和互信息当$\tau=65$时下降到接近0。换句话说，我们得到一种提示：一到大约每16s，就会有“一些有意

思的事情”发生（频率为4Hz，共计65步，每步所需时间为16.25s）。

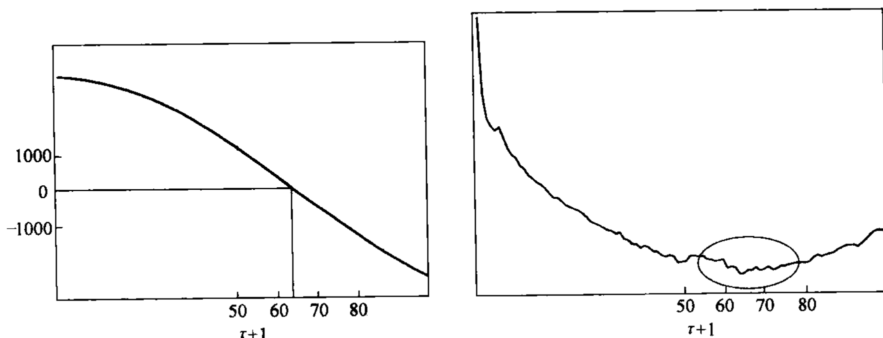


图6.3 图6.2中显示的 $x(t)$ 的自相关性（左图）和互信息（右图）
（自相关性下降到大约为0，互信息有一个延迟为 $\tau=65$ 的极小值）

通过 $x(t)$ 的频谱（见图6.4）可以看到数据中有两个相关的周期，一个周期每248个数据点重复一次，令一个较小周期每84个数据点重复一次。这粗略地等于机器人从场地的一个角落运动到下一个角落的时间（$84\times0.25s=21s$）和运动完一圈（$248\times0.25s=62s$）的时间。

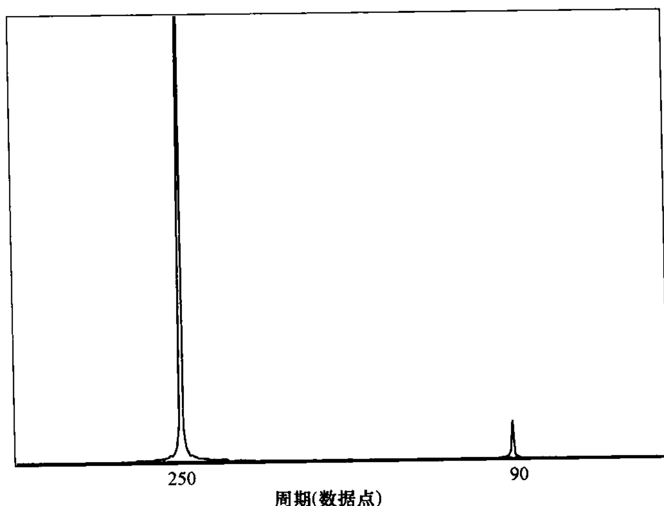


图6.4 $x(t)$ 的频谱

在这个例子中，我们得到了数据与物理考虑的一致信息。大约每20s机器人转过一个角，大约每60s机器人走完一圈。自相关性和互信息都表明了相关的新数据点每16s出现一次，粗略地等于通过两个拐角之间耗用的时间。

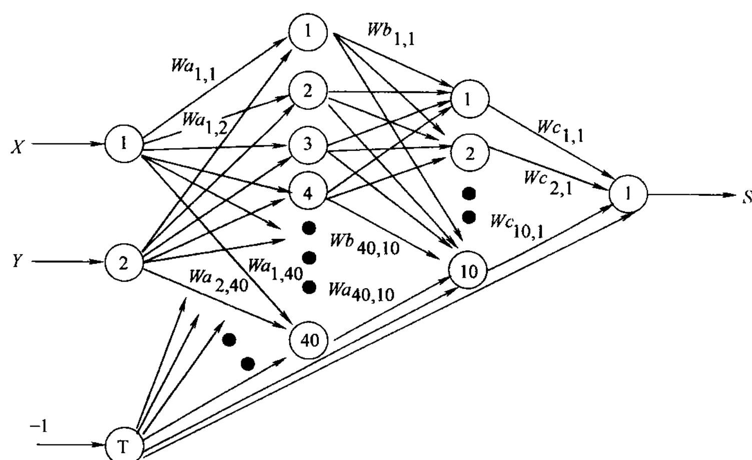
通常移动机器人学应用中一个可以接受的采样频率是大约每周期10个样本。在这个例子中最短的周期大概是20s。这表示应该每2s记录一次机器人的

位置，采样频率为每250ms一个样本，这就意味着需要以1:8的速度来降低采样频率。

6.3 案例研究：人工神经网络模型获取

这个案例研究介绍一种方法，它可以通过探索学习位置感知的映射，可以预测那些从没有经过的位置传感器的读数。采用一个多层感知器来实现这个功能。

图6.5显示了两层网络结构。它是一个多层感知器，将机器人在（x，y）坐标系中的当前位置和一个用于建模的超声波传感器（声纳）的距离读数范围联系起来。使用了16个网络，每一个用于一个Nomad 200移动机器人的一个超声波传感器。



T: 阈值单位

W: 权重

S: 机器人的超声波传感器数据

X: 机器人在笛卡儿系统下的x坐标

Y: 机器人在笛卡儿系统下的y坐标

第一个隐藏层单位: 40

第二个隐藏层单位: f0

输入单位: 2

输出单位: 1

阈值单位: 51

阈值数值: 1

总权重: 490

图6.5　用于获得Nomad 200移动机器人超声波传感器模型的多层感知器

6.3.1 实验步骤

为了取得训练数据，移动机器人有序地、有规律地通过目标环境，在有规

律的时间间隔下获得超声波传感器读数。记录下从机器人的里程表和超声波系统中得到的（x，y）坐标，用于之后的离线网络训练。为减少测程偏移带来的误差，机器人的轮子编码需要频繁校准，而且机器人的路径选择为直线，而不是曲线，这样带来测程误差较小。图 6.6 为两种实验方案，显示了机器人为获得训练数据所经过的不同路径。

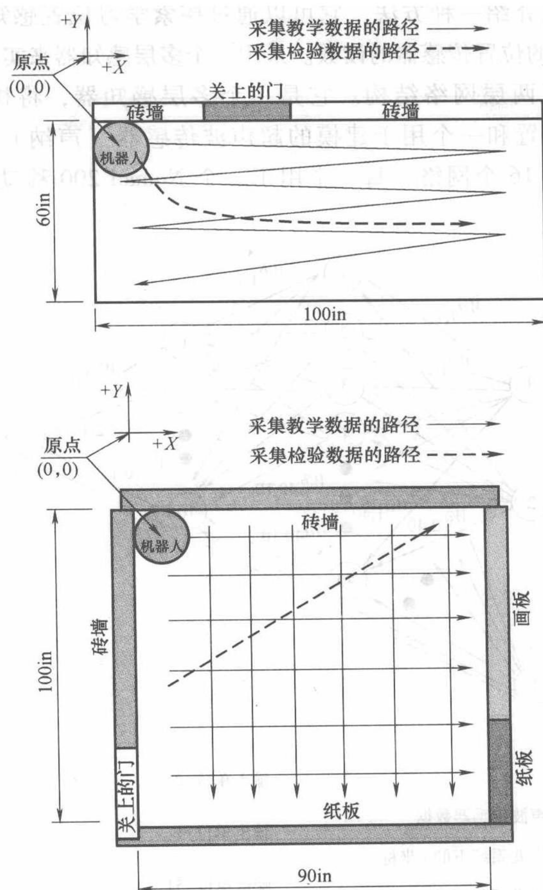


图 6.6　在两种不同环境下进行数据采集的实验方案

然后机器人将会按照一条不同、但更有趣的路径来采集检验数据。正如在图 6.6 里面看到的那样，训练和检验路径只在很少的几个点处重合。如果所获得的模型对机器人与这两种环境的交互作用有一个普遍的解释力，那么人工神经网络沿检验路径预测的感知与机器人的实际感知作比较时，这种解释力就会表现出来。

6.3.2　实验结果

1. 预测传感器的感知

图 6.7 为图 6.6（上图）中检验路径对应的预测和实际传感器读数的关系曲线。从中可以看出，因为光滑木制门的镜面反射，在样本数为 20 时，网络能够预测出超声波测距读数的突然增长。为便于比较一个简化的数值模型（Nomad 仿真器）给出的（错误）预测也在图中示出。

同样，图 6.8 为沿着图 6.6（下图）的检验路径的习得模型对于机器人感知的预测。

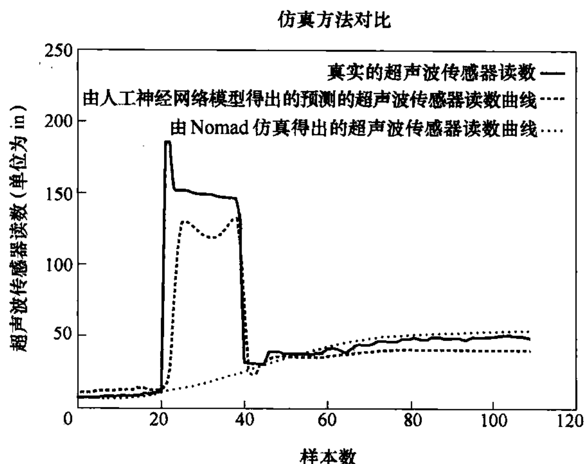


图 6.7　在墙壁沿行过程中超声波测距读数的建模，在样本号 20 处的突然增长是由于一扇木制门的镜面反射（一般的建模方法不能预测出来，因为这些模型假定在整个环境中的表面结构是一致的）

同样，所得的模型也能预测出在（400，-400）的位置处距离记录的突然下降。

2. 预测机器人行为

到目前为止，已经有迹象表明，人工神经网络模型能够预测出机器人在目标环境中的超声波传感器读数。这显然是十分有用的，而我们更感兴趣的是能预测出机器人在目标环境中执行一个特殊控制程序时的行为。

例如，使用一个电路控制程序（如一个不需要学习的有固定控制结构的程序），来实现墙壁沿行行为，并在真正的机器人、人工神经网络仿真，以及简单的数值仿真中执行这个程序。

结果是惊人的！图 6.6（上图）环境中，由于大门的镜面反射，实际上机器人冲向了门，错以为前方有更多空间，如图 6.9 所示。

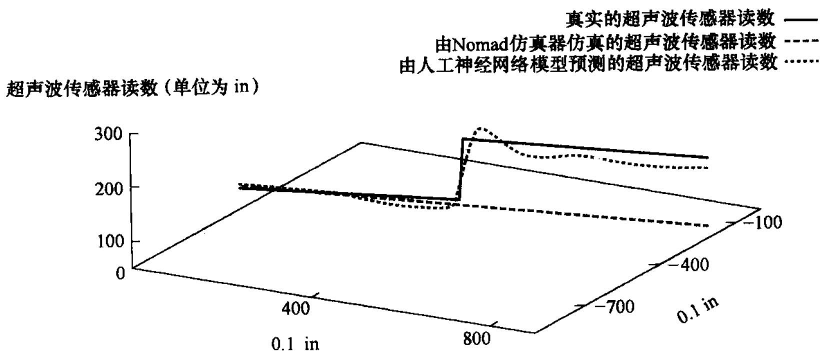


图 6.8　人工神经网络模型对检验数据的响应（一般的仿真器无法预测到由于镜面反射而引起的距离读数的突然增长，而学习型仿真器却能正确地预测到）

因为简单的数值模型假设在整个环境中表面结构均匀，所以它不能预测碰撞（ Nomad 仿真器见图 6.9 ），而人工神经网络仿真器认为这是所有结果。图 6.9 是对事实的一个说明，即谈论机器人的行为及其仿真之间这些不是轻微差别：这些都是重大差异，导致品质上完全不同的行为！

运行另一个简单的电路控制程序，这次是在图 6.6（下图）显示的环境中。该程序现在是一个寻找自由空间的程序，程序首先读取机器人所有 16 个超声波传感器读数，然后在最大读数方向上移动 1in（2.54cm），然后重复此过程，直到要么已过了 100in 的距离，要么机器人的红外线传感器检测到了障碍。这是一个充满危险的计划，因为即使是轻微的偏差也将让机器人进入环境中的不同区域，导致一个完全不同的轨迹。结果如图 6.10 所示。

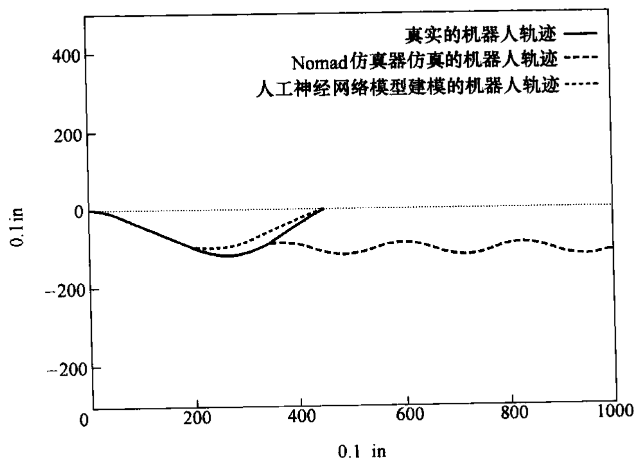


图 6.9　因为假设墙面均匀，通用仿真器无法预测真正的机器人遇到的由镜面反射引起的碰撞（该学习型网络仿真器正确预测了碰撞）

在一个统一的环境中，人们会期望一个寻找自由空间的程序能使该机器人走向环境的几何中心，使机器人围绕中心游走，这正是简单的数值模型所预测的。图6.10即为使用该程序获得的仿真轨迹与实际机器人运动轨迹的比较。

然而，实际上机器人移向了环境的边缘，这正是人工神经网络仿真器的预测。

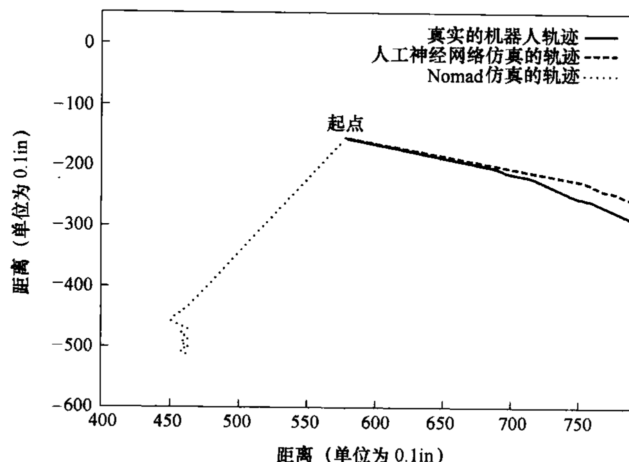


图6.10 对应于寻找自由的空间程序的仿真轨迹和实际机器人运动轨迹［简单的数值模型预测，机器人将移动到该机器人所处的环境的几何中心（位于（450，-450））。人工神经网络仿真器预测的轨迹非常接近实际上采取的轨迹］

3. 学习控制器的行为预测

目前为止，用来预测机器人行为的控制程序仍是相当简单的电路程序。这些程序取传感器读数为输入，并执行一个具体的、用户定义的行动作为反馈。

在这些实验中，机器人的行为主要由两个部分构成：机器人的感知和控制策略使用。任何仿真误差，仅当它使用一个电路控制程序时会影响机器人一次，如感知。控制程序是用户提供并且固定的，因此不会受仿真误差的影响。

另一方面来说，如果使用一个学习控制器，任何由仿真误差产生的问题会更为严重。首先，机器人将在错误的感知的基础上学习控制策略，然后它会执行错误的控制策略，同时读取错误的传感器数据作为输入。仿真误差在这些情形下有双重影响，因此，使用学习控制器的实验可以很好地对“仿真器到底有多可靠”进行敏感的测量。

4. 实验设置

我们进行一个具有本能规则的学习控制器（该控制器类似于［Nehmzow，2003a，p. 76ff］中所描述）的实验，实验中使用一种模式联想器。因此，控制

策略按照模式联想器的网络权重被编码。学习过程的目的是得到墙壁沿行行为。

无论是在简单的数值模型中，还是在训练网络模型中，都是在仿真中进行学习的。然后取得训练网络的权数，并加载到真正的机器人模式联想器，以控制机器人的运动。

真正的机器人轨迹被绘制在各仿真器预测的轨迹上。这些轨迹显示如图6.11所示。

从图6.11可以看出，人工神经网络仿真器的表现优于简单的数值机器人仿真器。不过，也可以看到，我们对于任何误差都产生双重影响的更敏感实验的假设为真：相对于图6.9和图6.10，这里预测轨迹和仿真轨迹彼此不是那么紧密了。

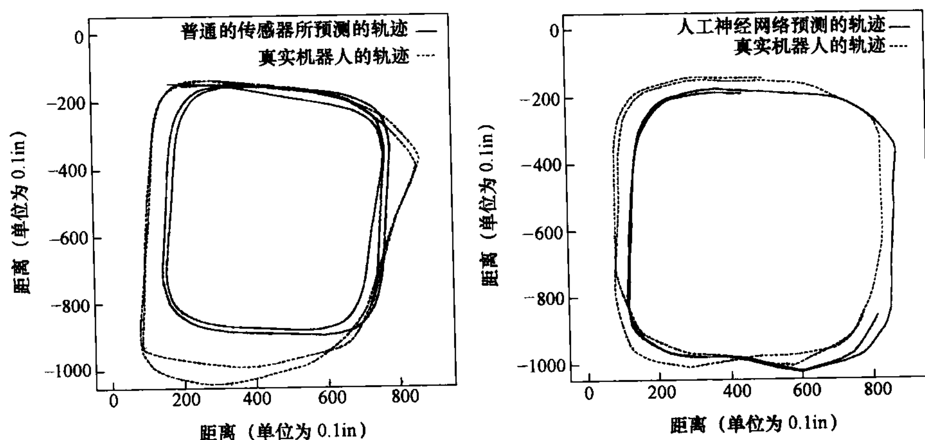


图6.11　左图为普通的仿真器所预测的机器人的轨迹和真正的机器人使用普通仿真器所产生的权数；右图为人工神经网络预测机器人轨迹类似真正的机器人使用网络仿真器所产生的权数

6.4　线性多项式模型和线性递推关系

6.4.1　导言

用真实的数据对输入变量 $u(t)$ 和输出变量 $y(t)$ 建模更有优势，即对模型化的行为而言，基于目标环境中运行的模型化智能体获得的真实世界数据的模型比基于一般假设的普通模型更可靠。上一节中给出的案例研究表明了这一点。

不过，使用人工神经网络对含糊的机器人、任务和环境的相互依存关系建

模得出一个含糊的模型。分析“到底什么正在发生”仍然是不可能的。为了做到这一点，需要一个解析数学模型。

一个取得解析模型的可行方法是将输入输出关系作为线性或非线性多项式进行建模，称为 ARMAX（具有外部输入的自回归滑动平均模型）或 NARMAX（非线性的 ARMAX）。

用这种易懂的多项式函数而非难懂的方法（如人工神经网络），对输入输出关系建模有相当大的优势：

1）输入输出的表达法非常紧凑，只需要很小的空间（内存）和处理时间来计算。

2）它们适合严格的数学分析。举例来说，机器人速度模型可以通过微分变成机器人加速度模型，加速度模型可以通过积分成为速度模型。此外，从有限的数学函数估计（如李雅普诺夫指数）或相关维数这样的参数比从时间序列估计更简单。

3）输入输出的关系可进行图形分析；图示是直接的，而对于难懂的模型，这是不可能的。

4）所获得的模型实际上表明了输入和输出之间的关系。参数和滞后表明了相关过程的特征。诸如“如果某个传感器失效了会发生什么事?”，“哪个传感器是最重要的（即将更昂贵的传感器装在哪里是最有效的)?”，或“如果环境以某种方式变化了会发生什么事?”这样的问题就可以得到处理。

5）通过各种方法得到对相似行为的分析。例如，通过来自控制理论的控制器与基于机器学习技术的控制器实现一个特定的机器人行为，有这些行为的模型用于分析时更容易：当一个易懂的数学表达式可供分析时，稳定性、噪声敏感度、相关和不相关传感器信号的辨识都变得更容易。

后面的章节将介绍获得这种易懂模型的两种途径，NARMAX 建模和 ARMAX 建模。这两种方法都将 $\boldsymbol{u}$ 和 $\boldsymbol{y}$ 之间的关系看作多项式表达，ARMAX 是线性的，而 NARMAX 是非线性多项式。两种情况下，模型都是易懂的，可以有系统地加以分析。

6.4.2　ARMAX 建模

ARMAX（具有外部输入的自回归滑动平均模型）是一个离散时间序列模型，普遍使用在系统辨识中，它将独立输入变量 $\boldsymbol{u}(t)$ 和非独立输出变量 $\boldsymbol{y}(t)$ 之间的关系作为式（6.1）给出的线性多项式：

$$y_t = -a_1y_{t-1} - a_2y_{t-2}\cdots - a_iy_{t-i} + b_1u_{t-1} + b_2u_{t-2}\cdots + b_iu_{t-i} + d_1e_{t-1} + d_2e_{t-2}\cdots + d_ie_{t-i} + e_t \tag{6.1}$$

输入 u，输出 y，噪声模型为 e，a_k、b_k 和 d_k 作为待定模型参数。过程如图 6.12 所示。

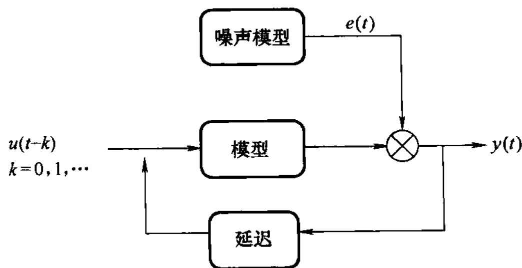


图 6.12　ARMAX 系统辨识过程

ARMAX 模型已广泛用于系统辨识研究，详细资料可在例如 [Pearson，1999，Box et al，1994] 中查阅。

ARMAX 模型有局限性，因为它是一个线性模型。然而，正如我们将看到的，对于许多机器人建模任务线性模型已足够，并且对诸如传感器感知—电动机响应的输入输出关系线性模型往往是可行的。

6.4.3　使用科学程序包的 ARMAX 模型

ARMAX 系统辨识是许多科学程序包（例如 Scilab 或 Matlab 的）的部分，不进行其他编程时也可用。后文会有一个例子说明在 Scilab 中如何用单行命令完成。下面列出的程序提供了一个对用户更加友好的 ARMAX 系统辨识过程的应用。

```
function[ypred,arc] = armaxid(y,u,r,s)
// (c) Ulrich Nehmzow
// 对 n 维输入和一维输出的具有外部输入的自回归滑动平均模型(ARMAX)辨析

// 程序用时间序列第一半创建模型,然后显示模型预测的整个时间序列

// u 是容量[(维数 n)(样本—容量)]的输入信号
// y 是容量[(维数 n)(样本—容量)]的输出信号
// r 是输出 y 的回归阶数
// s 是输入 u 的回归阶数

// ypred 是预测模型的输出
// arc 是得到的 ARMAX 模型

[a b] = size(y)
if (a > 1)
        printf("Only one - dimensional outputs permitted!  - Program aborted\n")
        abort
end
```

```
[ny,samples] = size(y)
[nu samples] = size(u)

// 检查两个时间序列一致的长度——否则正确
if (samples/2 - int(samples/2) ~ =0)
        y(samples + 1) = y(samples)
        u( : ,samples + 1) = u( : ,samples)
end

[ny,samples] = size(y)
[nu samples] = size(u)

modellength = int(samples/2)

// 加入第二(空的)输入线进行 armax 常规工作
y = ([y;zeros(1:samples)])

// 用数据第一部分进行 ARMAX 辨识
[arc, la, lb, sig, resid] = armax(r,s,y( : ,1:modellength),u( : ,(1:modellength)))
disp(arc)

// 现在对数据第二部分计算预测模型的值
ypred = y(1,1:modellength)

for i = modellength + 1:2 * modellength
        ypred(i) =0
        // 加入输出相关的成分
        for lag = 1:r
//              printf("Output component % f\n",la(1,2 * lag + 1))
                ypred(i) = ypred(i) - la(1,2 * lag + 1) * ypred(i - lag)
        end
        // 加入输入相关的成分
        for lag = 0:s
//              printf("Lag % d\n", lag)
                for inp = 1:nu
//              printf("Input % d\n", inp)
                ypred(i) = ypred(i) + lb(1,nu * lag + inp) * u(inp,i - lag)
//              printf("Adding % f x % f\n", lb(1,nu * lag + inp),u(inp,i - lag))
```

```
            end
        end
end

// 现在绘制预测模型的输出和实际的输出
xset("auto clear","off")
xbasc( )
plot2d([modellength:samples], ypred(modellength: samples),3,
   rect = [modellength,min(min(y),min(ypred)) -0.1, samples,
   max(max(y), max(ypred)) +0.1])
plot2d([modellength:samples], y(1,(modellength:samples)),5,
   rect = [modellength,min(min(y),min(ypred)) -0.1,samples,
   max(max(y),max(ypred)) +0.1])
xtitle('','Data point','')
legends(['Original output''Model - predicted'],[5 3],3)
printf("Estimated standard deviation of noise and residual: % f\n",sig(1,1))
printf("Sum squared error: % f\n", sqrt(sum((ypred - y(1,:))^2)))

// 现在计算如果个体输入成分被重置为零时的误差
ypred = y(1,1:modellength)

for block = 1:nu
        for i = modellength + 1:2 * modellength
                ypred(i) =0
                // 加入输出相关的成分
                for lag = 1:r
                        ypred(i) = ypred(i) - 1a(1,2 * lag + 1) * ypred(i - lag)
                end
                // 输入相关的成分
                for lag = 0:s
                        for inp = 1:nu
                            if(inp ~ = block)
                              ypred(i) = ypred(i) + 1b(1,nu * lag + inp) * u(inp,i - lag)
                            end
                        end
                end
        end
        printf("Blocking input %3d: error = %6.1f\n",block,norm(ypred - y(1,:),2))
```

```
        sse(block) = norm(ypred - y(1,:),2)
end
// 现在按有利于出版的格式打印

// 打印输入相关的成分
printf("b Matrix\n")
for lag = 0:s
        printf("   t - %d   ",lag)
end
printf("SSE\n")
for inp = 1:nu
        for lag = 0:s
                printf("%9.2f", 1b(1,nu * lag + inp))
        end
        prinft("%9.1 f\n", sse(inp))
end

// 打印输出相关的成分
printf("\na Matrix\n")
for lag = 1:r
        printf(" y(t - %d) ", lag)
end
printf("\n")
for lag = 1:r
        printf("%9.2 f", 1a(1.2 * lag + 1))
end
printf("\n")
```

例：ARMAX 模型，应用科学程序包

下面的例子演示了如何使用 Scilab 得到线性多项式模型。

假定有满足式（6.2）关系的数据：

$$y(t) = 0.5u(t) - 0.3u(t-1) + 1.5u(t-2) - 0.7y(t-1) \tag{6.2}$$

首先，生成一个随机输入向量 $\boldsymbol{u}$：

u = rand(1: 100);

现在计算输出变量 y：

y(1) =0.2

y(2) =0.2

for i =3:100

```
        y(i) =0.5*u(i) -0.3*u(i-1) +1.5*u(i-2) -0.7*y(i-1);
    end
```

式（6.2）中的回归阶数是 y 为 1，u 为 2。以下 Scilab 命令将确定描述数据的 ARMAX 模型：

```
Armax (1, 2, [y'; zeros (1:100)], u)
```

得到的结果为：

```
        A(z^-1)y = B(z^-1)u + D(z^-1)e(t)
        A(x) =
!       1 +0.7x       1.329E-16x   !
!                                  !
!       0             1            !
B(x) =
!                               2  !
!       0.5 -0.3x +1.5x            !
!                                  !
!       0                          !
```

这个结果等同于式（6.3）：

$$y(t) + 0.7y(t-1) = 0.5u(t) - 0.3u(t-1) + 1.5u(t-2) + \text{噪声} \tag{6.3}$$

式（6.3）变形为：$y(t) = 0.5u - 0.3u(t-1) + 1.5u(t-2) - 0.7y(t-1) +$ 噪声，即式（6.2）中表达的关系。

6.5 NARMAX 建模方法

上文所述的 ARMAX 方法在确定 $y-\boldsymbol{u}$ 的输入输出关系的简明模型上有很大优势，模型可以被分析和解释。然而，它的缺点是 ARMAX 只能构建线性的输入输出关系。这对于一些在机器人学上的应用足够了，但并非所有。这就是为什么要引进 NARMAX（非线性 ARMAX）。

NARMAX 方法是一种参数估计方法，确定重要的建模条件及未知的非线性动力系统的参数。对于单输入单输出系统，这个模型采用式（6.4）的形式：

$$\begin{aligned} y(k) = F[&y(k-1), y(k-2), \cdots, y(k-n_y), \\ &u(k-d), \cdots, u(k-d-n_u), \\ &e(k-1), \cdots, e(k-n_e)] + e(k), \end{aligned} \tag{6.4}$$

式中，$y(k)$，$u(k)$，$e(k)$ 分别为采样输出、输入和无法观察的噪声序列；n_y，n_u，n_e 为阶数；d 为时间延迟，F [ ] 为一种非线性函数，通常被用作论据的一

个多项式或小波多分辨率的展开式。

通常只有输入和输出的测量值 $u(k)$ 和 $y(k)$ 是可用的，并且研究者必须处理这些信号来估计系统模型。

NARMAX 方法解决这个问题分为以下几个步骤：

1）结构检测；

2）参数估计；

3）模型验证；

4）预测；

5）分析。

这些步骤形成一个估计工具包，允许用户建立一个简洁的对该系统的数学描述 [Chen and Billings，1989]。程序由确定结构或重要的模型项开始，接下来继续估计模型参数。这些程序现在已经很好地建立并应用于很多建模领域 [chen Billings，1989]。一旦模型的结构被确定，就可以估计模型的那些未知参数了。如果要获得正确的参数估计，几乎总是观察不到的噪声序列 $e(k)$ 就必须根据模型进行估计和调整。模型验证方法用于确定该模型是否是恰当的。一旦模型被接受，它可以用于预测系统对于不同输入的输出以及研究所试验系统的特征。

讨论怎样获得一个 NARMAX 模型是超出本书范围的，但是在很多文献中都有介绍（见特别是 [Chen and Billings，1989]）。

现在将转向系统辨识技术在机器人学（“机器人辨识”）方面的应用。特别地，我们要演示机器人辨识怎样如实而精确地对一个移动机器人在一个特定的环境下的运行（环境辨识）进行仿真；如何在实际上不写机器人代码的情况下促进跨平台的编程（任务辨识）；如何用于把一种传感器的形式“翻译”成另一个以及允许为用一种传感器的机器人编写的代码在一个没有这种传感器的机器人上执行（传感器辨识）。

6.6 精确仿真：环境辨识

6.6.1 导言

我们在环境辨识中的目标是获得精确易懂的可以用于代码开发的机器人—环境交互作用的计算机模型。普通的仿真程序被具体的机器人环境交互模型替代，来源于机器人学实验的真实世界获得的数据。

这一节解释我们获得环境模型的过程。这里用一个简单的机器人行为来阐述主要的机制。图 6.13 展示了在第一个例子中讨论的实验研究的建模关系。

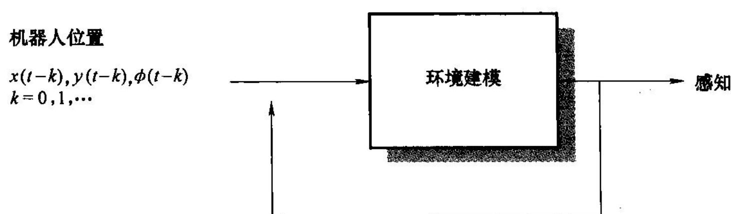


图 6.13 环境建模：已知的函数（如表 6.1 给出的多项式）
绘制机器人位置到感知的映射

在这个例子中，我们选择研究一个麦哲伦专业版移动机器人的墙壁沿行行为（实际上驱动机器人的控制代码不是墙壁沿行的而是一个避障的程序。然而，机器人在与环境的相互作用中产生了墙壁沿行轨迹）。使用的机器人为图 1.1 中的麦哲伦专业版，图 6.14 所示为用顶部摄像机以每 250ms 拍摄一次形成的轨迹。

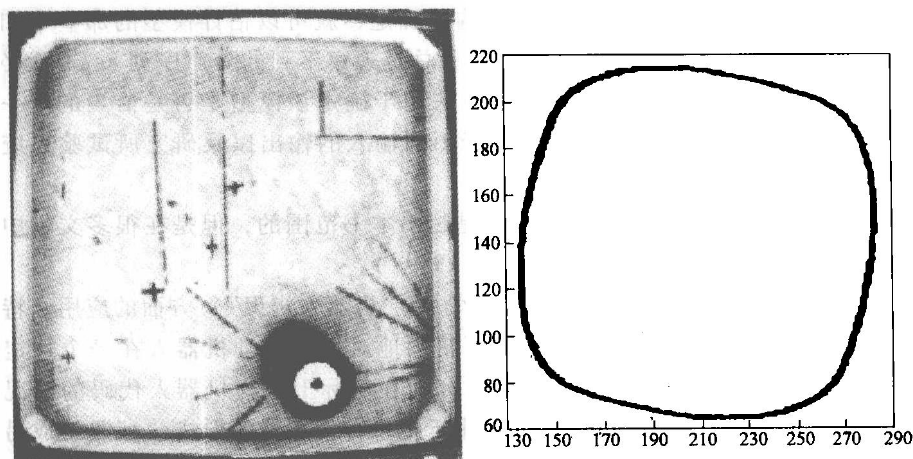


图 6.14 实验进行的环境（左图），机器人轨迹（右图）
（机器人出现在左图底部的右下角）

首先，这个数据随后以 1:15 的比例被二次采样，这样在数据点之间时间为 3.75s。在这个实验中机器人的平均速度为 8cm/s，经过的机器人位置之间的距离大约为 30cm。

为了获得非线性模型，我们采用 NARMAX 模型辨识方法。首先，该模型结构由选择输入输出的回归阶数和次数决定。“次数”的定义为在一项中所有指数的总和，“项”是一个数学表达，如表 6.1 中的每一行所示。

为了确定一个适当的模型结构，我们使用 [Korenberg et al，1988] 中介绍的正交参数估计算法。这指出（在对模型的计算前）哪些模型项对输出的计算的影响是显著的。

然后，我们用可用数据（训练数据）的前一半得到模型，用剩下的一半（验证数据）验证它。

结果的模型列于表6.1中，它计算由机器人（L67见图6.15）左侧67°的激光传感器测量到的距离作为它自身位置(x,y)的函数。

表6.1 建模机器人单射线激光传感器L67作为机器人位置(x,y)的函数的多项式的参数（模型的时间序列如图6.16所示，也可见图6.15）

```
L67(t) = +1.8801351
         +0.0087641 * x(t)
         -0.0116923 * x(t-1)
         -0.0060061 * x(t-2)
         +0.0116420 * y(t)
         +0.0143721 * y(t-1)
         -0.0064808 * y(t-2)
         +0.0004983 * x(t)^2
         +0.0021232 * x(t-1)^2
         +0.0006722 * x(t-2)^2
         -0.0002464 * y(t)^2
         +0.0018295 * y(t-1)^2
         +0.0015442 * y(t-2)^2
         -0.0028887 * x(t) * x(t-1)
         +0.0023524 * x(t) * x(t-2)
         +0.0002199 * x(t) * y(t)
         -0.0025234 * x(t) * y(t-1)
         +0.0022859 * x(t) * y(t-2)
         -0.0029213 * x(t-1) * x(t-2)
         +0.0006455 * x(t-1) * y(t)
         +0.0014447 * x(t-1) * y(t-1)
         -0.0027139 * x(t-1) * y(t-2)
         -0.0004945 * x(t-2) * y(t)
         +0.0003262 * x(t-2) * y(t-1)
         +0.0009349 * x(t-2) * y(t-2)
         -0.0010366 * y(t) * y(t-1)
         +0.0013326 * y(t) * y(t-2)
         -0.0037855 * y(t-1) * y(t-2)
```

对于真实激光传感器和由表6.1中模型预测的比较，如图6.16所示。它清楚地表示出机器人的传感器L67（t）确实可以简单精确地用时间t到$t-1$间的机器人位置(x,y)的函数所建模。建模和真实的数据的皮尔森（Pearson）相关系数为0.75（显著的，$p<0.01$）。

注意到为了对激光传感器的感知建模，机器人的方向ϕ是不需要的，这是

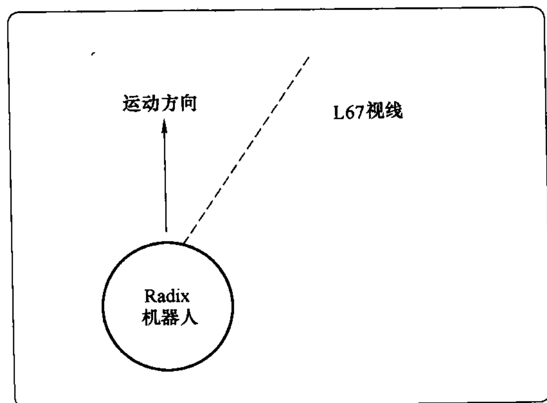


图 6.15 L67(表 6.1 中建模的)是接近机器人右侧的单射线激光传感器

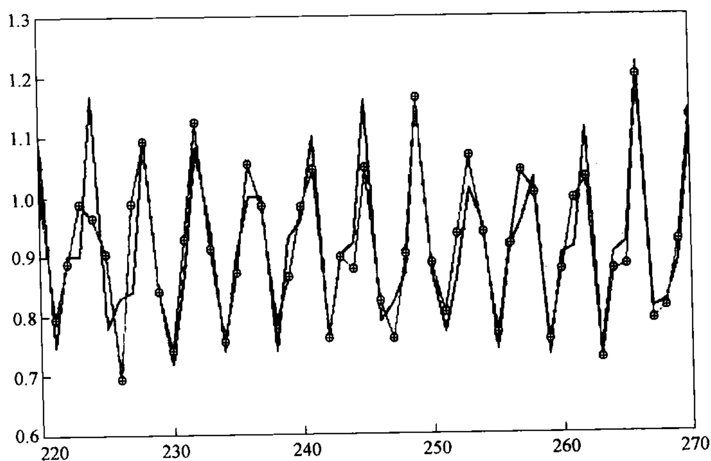


图 6.16 建模机器人的激光传感器 L67 为位置的函数（参见表 6.1）

注：真实的激光传感器数据表示为带圆圈的线，预测模型的输出为不带圆圈的线。

很有趣的。原因是机器人的运动是有限制的运动（沿着环境的周界），机器人通过确定 (x,y) 实质上已经确定了方向 ϕ，因此 ϕ 不需要在模型中明确。

6.6.2 环境建模：ARMAX 例子

简单的举例中，环境的模型不必一定是非线性的，接下来我们要建立一个线性的激光感知的 ARMAX 模型，一个麦哲伦专业版模型，它的轨迹和图 6.14 所示非常接近。

当机器人沿着如图 6.14 所示的轨迹运动，由激光测距仪获得的深度数据每 250ms 记录一次，机器人的位置为 (x,y) 和旋转速度为 $\dot{\phi}$ 。这个数据随后以1:15

的比例被二次采样（二次采样的讨论见6.2.1节），所有在“偏左45°”和“水平直线”之间的激光传感器数值被平均，从而获得图6.17所示的激光传感器数据 $LM(t)$。机器人的位置 (x,y) 和旋转速度 $\dot{\phi}$ 也见图6.17。

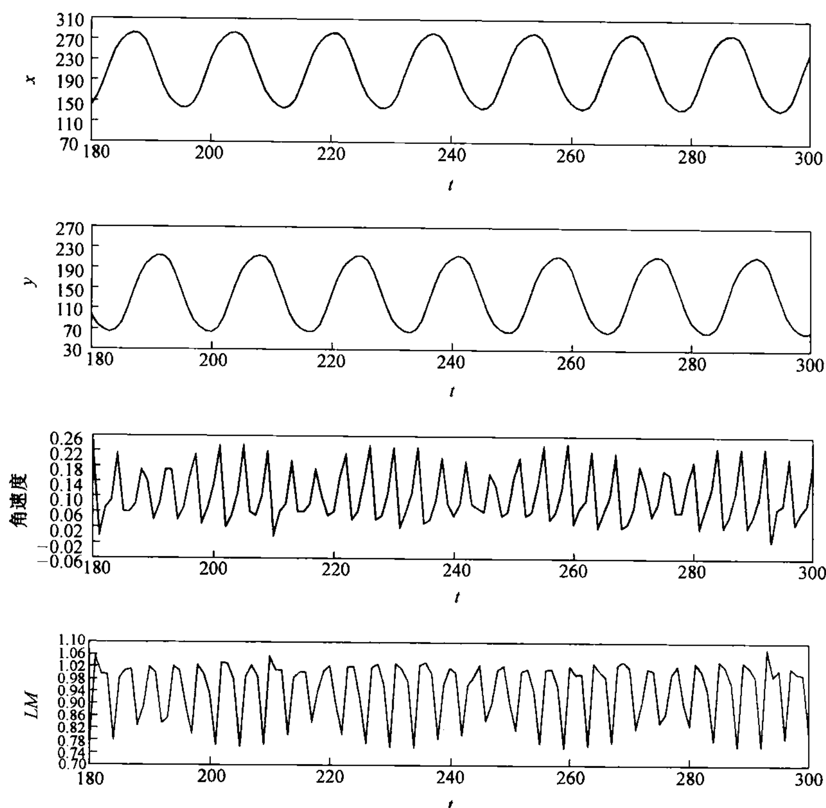


图6.17 用于环境辨识的机器人位置（上面的三个曲线图）和激光传感器数据（最后一个曲线图）

我们现在用Scilab ARMAX包获得ARMAX模型。

$LM(t)=f(x(t),y(t),\dot{\phi}(t),x(t-1),y(t-1),\dot{\phi}(t-1))$，我们要用前500个数据点建立模型（模型数据），用剩下的451个数据点验证模型（验证数据）。

```
armax(0,1,[y(1:500),zeros(1:500)']',
        [u(1:500,1),u(1:500,2),u(1:500,3)]')
ans = A(z^-1)y = B(z^-1)u + D(z^-1)e(t)


A(x) =
!   1     0  !
```

! 0 1 !

B(x) =

0.0078763 -0.0041865x

-0.0078344 +0.0101931x

-1.3727936 -0.1602578x

D(x) =

! 1 0 !

! 0 1 !

e(t) = sig * w(t) ; w(t) 2 - dim white noise

|0.0558112 0|

Sig =|0 0|

这个结果在模型中由式(6.5)给出：

$$
\begin{aligned}
LM(t) = & 0.0078763x(t) - 0.0041865x(t-1) \\
& - 0.0078344y(t) + 0.0101931y(t-1) \\
& - 1.3727936\dot{\phi}(t) - 0.1602578\dot{\phi}(t-1)
\end{aligned}
\tag{6.5}
$$

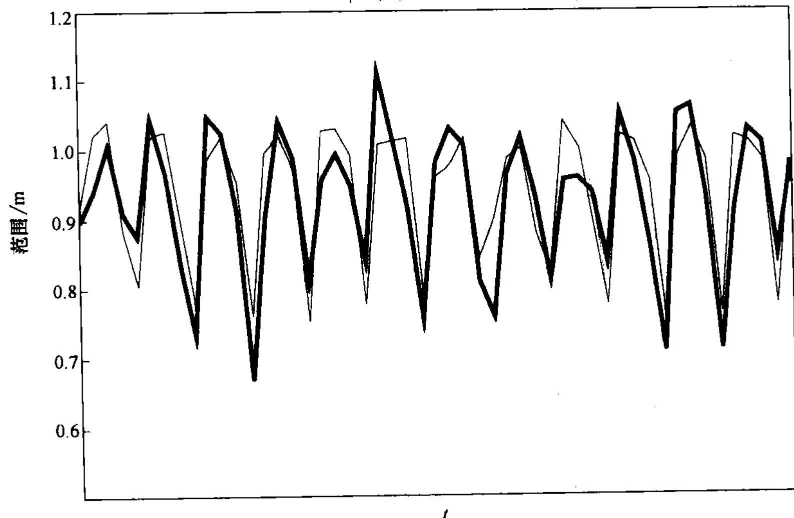


图 6.18 实际的激光传感器数据（细虚线）和预测模型激光传感器数据（式 6.5，粗实线）的比较

利用这一模型去预测我们验证数据（数据点 501 到 941）的激光传感器数据，就得到了如图 6.18 显示的结果。

6.6.3　通过环境建模进行定位

图 6.13 描述的建模情形是可逆的，并且能够用于自我定位。如果机器人的位置和在该位置获得的感知之间的关系已知，那么逆向过程也正确（假设这种关系由一个逆函数表达出来），并且机器人的位置可以由感知所决定。如图 6.19 所示。

1. 模型 1

我们将会通过一个例子来说明这个过程。

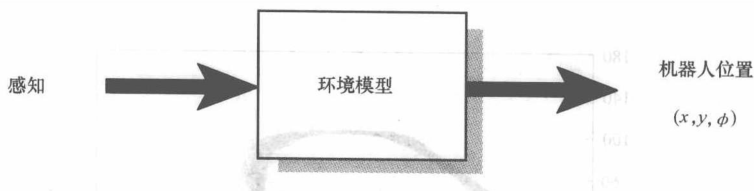


图 6.19　基于传感系统的定位

实验在一个大小大约 2m×2m 的封闭场地里进行，在这个场地里机器人做墙壁沿行运动。用顶部摄像机，每隔 6.25s 对机器人的动作和所有的感知进行一次记录。图 6.20 给出了该场地的鸟瞰图，在该图中机器人也清晰可见。

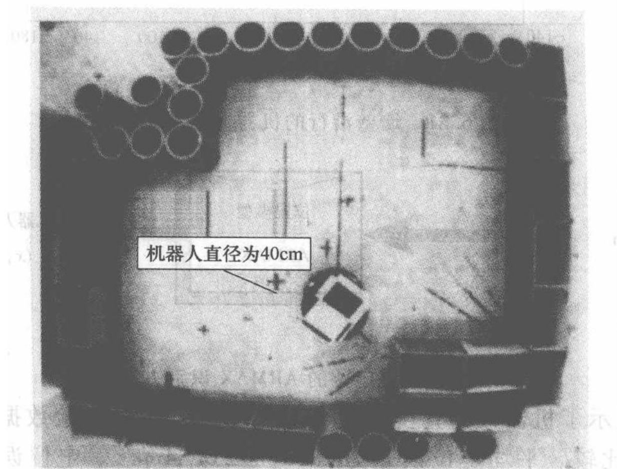


图 6.20　封闭场地的鸟瞰图

如图 6.21 所示，获得的机器人运动轨迹——通过顶部摄像机获得的地面实况。

这个实验的目的是为了利用从机器人的16个超声波传感器和12个激光传感器获得的信息，建立机器人的位置坐标（x，y）。然后利用一个ARMAX建模过程（图6.22中所示的）获得位置坐标（x，y）模型，给出如图6.22所示的预测的激光和超声波传感器数据。

利用在6.4.3节提到的程序，我们能够分别获得在表6.2和表6.3给出的$x(t)$和$y(t)$的模型。

尽管模型需要一个相对高级的回归程序，让机器人在我们的实验空间只通过传感器信息来实现自我定位，不过这还是可能的。图6.23显示了真实的(x，y)和模型预测出的（$\tilde{x}$，$\tilde{y}$）的比较。x（t）和$\tilde{x}$（t）的斯皮尔曼相关系数r_x，y（t）和$\tilde{y}$（t）的斯皮尔曼相关系数r_y，分别为0.96和0.95（显著的，$p<0.05$）。

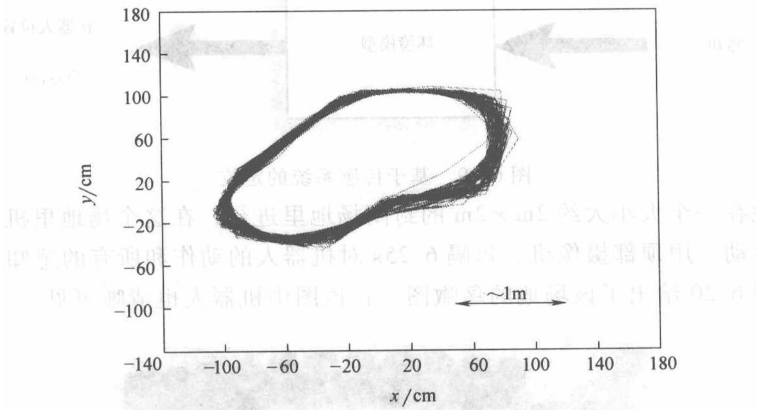


图6.21 墙壁沿行的机器人运动轨迹



图6.22 自定位的ARMAX识别方法

图6.24显示了机器人的真实运动轨迹和机器人通过检验数据进行建模得出的轨迹之间的比较。平均定位误差为29.5cm±0.84cm[⊖]。定位误差的分布如图6.25所示。

2. 模型2：模型1的改进

⊖ 作为对比，机器人的直径是40cm。

运用多数对机器人有用的传感器信息获得模型 1($x(t)$,$y(t)$)的目的，是为了研究这些信息对于定位是不是都需要。我们可以用表 6.2 和 6.3 证明并非如此：表中的最右边一栏给出了当某一个特定的建模条件改变时得到的误差平方和。六个对模型作用最大的传感器信号占了对模型作用的绝大部分。而有趣的是，在顶部附近任何地方都没有超声波信号；它们很明显包含了太多的不精确的互相矛盾的信息，这些信息在定位过程中同激光传感器数据一样重要。

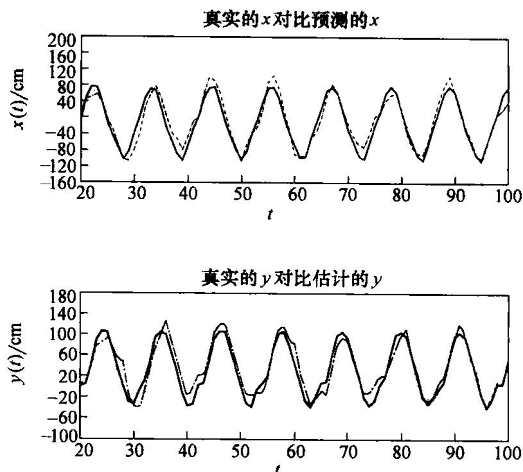


图 6.23　机器人的真实位置($x(t)$,$y(t)$)(粗实线)与利用表 6.2 和 6.3 给出的模型 1 中传感器感知估计的位置（虚线）的比较

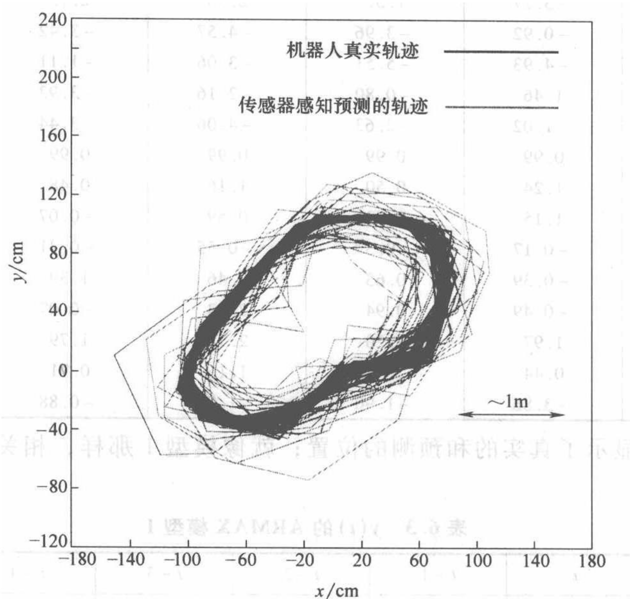


图 6.24　机器人真实轨迹（粗实线）与利用表 6.2 和 6.3 给出的模型 1 中超声波和激光传感器数据估计的轨迹（虚线）的比较

基于这种考虑，我们仅用六个最重要的传感器信号得出了模型 1 的改进。第二个模型如表 6.4 和表 6.5 所示。从这些表中我们看出定位只用较少的传感器信息就足够了，但是现在需要更高的回归阶数：机器人用了较少的传感器信息，

但耗费了更长的时间。一个 13 个样本的时间窗等同于超过 80s 的运动！

表 6.2 $x(t)$ 的 ARMAX 模型 1(例如，$x(t)$ = -47.40,*Laser*15(t) -89.78,*Laser*15($t-1$)…)

(SSE 为相应的项从模型中移除后的误差平方和)

	t	$t-1$	$t-2$	$t-3$	SSE
*Laser*15	-47.40	-89.78	-72.11	-21.42	**2506.4**
*Laser*30	-12.04	-14.14	-7.05	12.05	410.0
*Laser*45	-17.19	-9.57	-10.05	-2.25	634.9
*Laser*60	-3.37	4.99	-2.78	7.01	337.9
*Laser*75	18.90	9.85	45.78	16.14	**1784.3**
*Laser*90	2.39	9.28	-7.01	-0.52	337.9
*Laser*105	-14.71	-17.80	-9.74	-12.08	**1435.6**
*Laser*120	23.16	19.02	18.91	-1.52	**1734.7**
*Laser*135	-18.79	1.65	6.90	19.91	555.0
*Laser*150	2.05	-6.87	-11.79	10.39	402.0
*Laser*165	13.44	9.04	14.84	-2.08	**1195.2**
*Laser*180	18.36	14.95	19.65	-5.86	**1597.4**
*Sonar*1	-4.17	-3.14	-2.45	0.76	444.1
*Sonar*2	-3.02	-1.93	-0.91	-1.19	474.0
*Sonar*3	-3.27	1.52	2.98	2.15	364.0
*Sonar*4	-0.92	-3.96	-4.57	-3.42	636.5
*Sonar*5	-4.93	-5.51	-3.06	-1.11	710.4
*Sonar*6	1.46	-0.80	-2.16	-3.92	418.0
*Sonar*7	-1.02	-2.63	-4.06	-3.44	622.5
*Sonar*8	0.99	0.99	0.99	0.99	428.8
*Sonar*9	1.24	0.50	1.16	0.48	346.5
*Sonar*10	1.15	2.72	0.59	-0.07	356.7
*Sonar*11	-0.17	-0.39	-0.55	-0.41	317.5
*Sonar*12	-0.39	0.63	0.46	1.39	322.5
*Sonar*13	-0.49	0.94	2.83	-0.07	331.3
*Sonar*14	1.97	2.20	2.78	1.79	374.3
*Sonar*15	0.44	2.47	1.16	0.71	357.2
*Sonar*16	-3.86	-1.90	-1.01	-0.88	422.4

图 6.26 显示了真实的和预测的位置；就像模型 1 那样，相关性是高度显著的。

表 6.3 $y(t)$ 的 ARMAX 模型 1

	t	$t-1$	$t-2$	$t-3$	$t-4$	SSE
*Laser*15	35.65	7.86	-38.68	-74.11	-38.72	**1223.4**
*Laser*30	8.65	-5.01	-7.89	-12.35	-3.82	378.9
*Laser*45	8.18	-9.80	-0.75	-3.61	1.90	295.0
*Laser*60	-5.96	-0.57	13.33	10.59	-6.89	317.6
*Laser*75	3.74	5.79	0.27	29.41	26.38	**1285.7**
*Laser*90	-11.34	1.64	1.91	-1.28	-1.10	358.7
*Laser*105	3.52	-3.97	-14.41	-8.67	-5.30	817.6

（续）

	t	$t-1$	$t-2$	$t-3$	$t-4$	SSE
*Laser*120	7.41	17.19	20.37	16.40	13.45	**2114.1**
*Laser*135	4.13	-7.20	-12.44	-9.44	9.32	623.9
*Laser*150	0.58	10.29	-2.45	-8.90	1.23	320.3
*Laser*165	-6.16	1.86	11.38	15.51	0.53	794.1
*Laser*180	-10.22	2.12	4.88	18.30	9.67	852.5
*Sonar*1	-1.64	-2.72	-1.97	-2.74	-1.15	455.6
*Sonar*2	-0.42	-1.85	1.14	1.18	-0.66	280.5
*Sonar*3	-1.98	-3.82	-0.68	0.99	0.72	353.7
*Sonar*4	2.50	1.30	0.37	-3.46	-4.25	349.6
*Sonar*5	0.16	-1.91	-2.42	-1.83	-1.61	446.0
*Sonar*6	2.67	1.77	0.41	-0.79	-1.54	307.3
*Sonar*7	2.12	1.56	-1.13	-2.55	-1.00	297.4
*Sonar*8	-2.93	-2.93	-2.93	-2.93	-2.93	**1069.0**
*Sonar*9	0.75	0.35	1.02	1.72	1.56	333.0
*Sonar*10	-1.07	-0.35	2.37	1.30	1.37	301.0
*Sonar*11	0.24	1.19	0.66	-0.86	0.73	282.2
*Sonar*12	-0.06	0.48	0.61	0.61	1.72	285.4
*Sonar*13	-1.29	0.14	1.48	3.80	2.89	318.9
*Sonar*14	-1.95	0.19	1.61	2.51	2.01	292.1
*Sonar*15	-0.77	0.15	0.96	1.11	0.58	284.3
*Sonar*16	-1.21	-2.46	-1.59	-1.48	-0.09	392.6

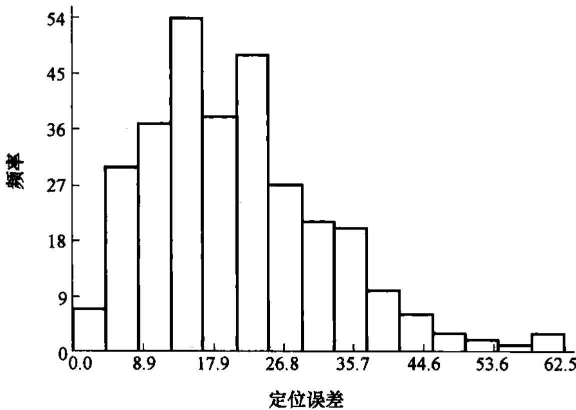


图 6.25　检验数据的定位误差的分布，用表 6.2 和表 6.3 中的模型 1

表 6.4　备选模型（模型 2）确定来自激光传感器信号 x

	$t-0$	$t-1$	$t-2$	$t-3$	$t-4$	$t-5$	$t-6$	SSE
*Laser*15	-55.94	-73.11	-75.63	-53.57	-5.18	44.67	67.42	1541.2
*Laser*75	-2.26	3.21	7.64	4.45	-2.06	-6.28	-12.83	388.3
*Laser*105	-13.16	-12.03	-8.71	-4.58	6.79	5.94	8.57	1388.3
*Laser*120	18.13	13.45	7.03	-7.22	-13.13	-18.91	-16.63	510.7
*Laser*165	11.64	15.93	13.81	9.17	5.46	0.70	-9.99	2133.5
*Laser*180	-3.53	-0.52	-2.58	-2.42	0.12	3.72	-1.11	354.3

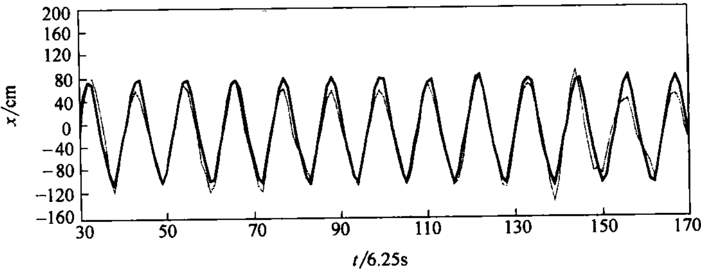
（续）

	t-7	t-8	t-9	t-10	t-11	t-12	SSE
*Laser*15	68.19	50.96	26.88	-13.36	-39.01	-58.16	1541.2
*Laser*75	-13.37	-12.57	-12.72	4.14	12.57	16.95	388.3
*Laser*105	5.79	-0.24	-5.02	-13.64	-13.53	-8.21	1388.3
*Laser*120	-6.17	5.74	10.95	8.52	6.19	5.97	510.7
*Laser*165	-7.16	-7.12	2.17	12.50	9.04	9.41	2133.5
*Laser*180	-5.95	-3.73	-0.24	1.46	7.79	12.13	354.3

表 6.5 备选模型（模型 2）确定来自激光传感器信号 y

	t-0	t-1	t-2	t-3	t-4	t-5	t-6	SSE
L. 15	12.36	-13.84	-39.92	-71.09	-64.45	-45.60	7.85	961.4
L. 75	0.51	0.05	-0.13	13.07	8.17	4.18	2.70	274.3
L. 105	6.92	-5.11	-9.54	-7.95	-8.41	-1.59	-0.17	784.1
L. 120	6.13	18.91	15.14	10.25	4.96	-0.88	-15.79	926.9
L. 165	-8.35	3.67	10.91	14.84	11.12	6.60	4.76	1245.9
L. 180	-4.24	-1.22	-1.57	-0.83	-0.78	0.06	0.61	263.8

	t-7	t-8	t-9	t-10	t-11	t-12	t-13	SSE
L. 15	37.98	47.64	61.94	48.53	12.55	-11.17	-41.20	961.4
L. 75	-9.25	-9.91	-11.62	-15.66	-5.29	6.28	10.47	274.3
L. 105	8.40	8.97	8.41	3.14	-9.05	-11.26	-10.78	784.1
L. 120	-14.63	-12.42	-5.81	4.24	8.46	6.59	6.44	926.9
L. 165	-6.42	-6.92	-10.99	-3.88	7.06	7.16	8.07	1245.9
L. 180	-0.27	-2.01	-2.64	0.20	-2.38	3.65	10.78	263.8



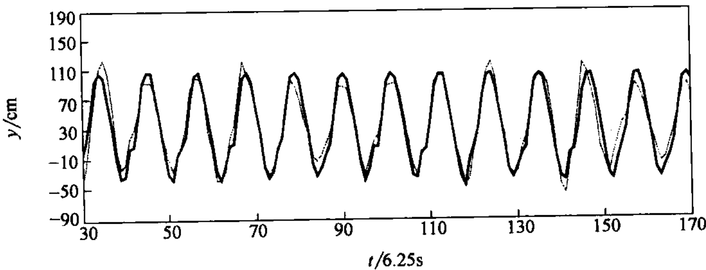


图 6.26 真实机器人位置 $(x(t), y(t))$（粗实线）和用表 6.4 和 6.5 中给出的备选模型 2 给出的传感器估计的位置（虚线）的比较

图 6.27 显示了实际的运动轨迹和模型预测的轨迹的比较。平均定位误差低于模型1:(22 ±0.7)cm。图 6.28 给出了误差分布情况。

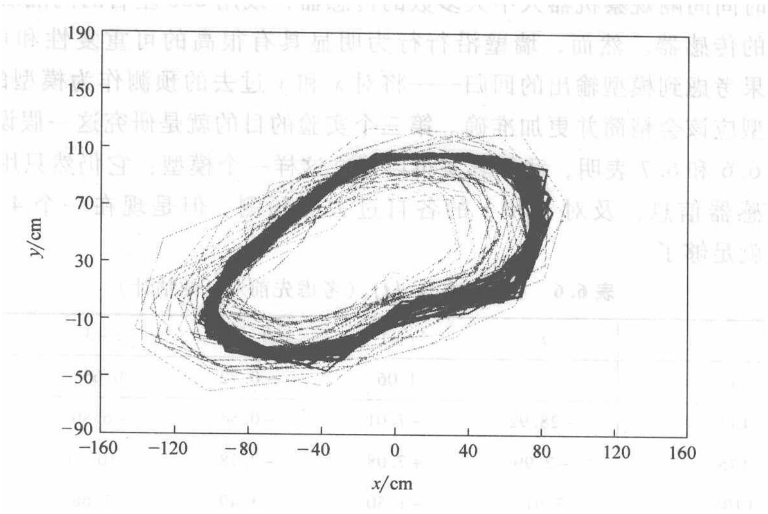


图 6.27 真实机器人轨迹(粗实线)和用表 6.4 和表 6.5 中给出的备选模型 2 给出的激光传感器估计的轨迹(虚线)的比较

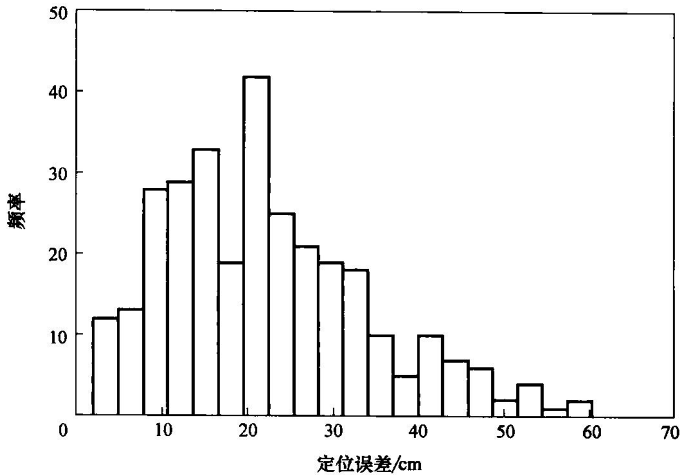


图 6.28 利用表 6.4 和表 6.5 给出的模型 2 中检验数据的定位误差分布情况

3. 模型 3：将运动考虑在内，模型 2 的进一步改进

模型 1 和模型 2 表明，要达到机器人直径一半的定位精度，可以通过用 24s 左右的时间间隔观察机器人中大多数的传感器，或用 80s 左右的间隔观察提供信息最多的传感器。然而，墙壁沿行行为明显具有很高的可重复性和可预测性，并且如果考虑到模型输出的回归——将对 x 和 y 过去的预测作为模型的一部分，那么模型应该会精简并更加准确。第三个实验的目的就是研究这一假设。

表 6.6 和 6.7 表明，我们其实可以建立这样一个模型：它仍然只用六个最重要的传感器信息，及对 x 和 y 的各自过去的预测。但是现在一个 4 次的回归（24s）就足够了。

表 6.6 模型 3 的 $x(t)$（考虑先前对 x 的估计）

	t	$t-1$	$t-2$	$t-3$	$t-4$
x		1.06	-0.72	0.65	-0.70
L15	-28.92	-7.01	-0.54	-6.80	
L75	-2.99	+7.08	-2.78	+10.33	
L105	3.91	-1.50	-3.49	+1.68	
L120	3.63	+1.28	-3.85	-0.24	
L135	-7.40	+2.14	+6.34	+0.01	
L180	2.59	+3.32	-8.06	+3.42	

表 6.7 模型 3 的 $y(t)$（考虑先前对 y 的估计）

	t	$t-1$	$t-2$	$t-3$	$t-4$
y		0.85	-0.18	0.09	-0.49
L15	9.96	-5.83	-15.84	-8.31	+0.88
L30	16.80	-9.71	-5.20	+2.48	+0.34
L75	1.46	+0.41	-5.68	+7.11	-0.48
L120	1.95	+2.97	+4.06	-4.00	+7.88
L135	1.99	+1.03	-5.28	+2.58	+5.50
L180	-2.62	+1.80	+4.10	-3.82	+0.34

图 6.29 所示的是在这种情况下真实轨迹和预测的轨迹。定位误差已经减少到近一半即（13 ±0.4）cm，相关系数 r_x 和 r_y 都是 0.98（显著性，$p<0.05$），定位误差分布如图 6.30 所示。

4. 模型 4：模型 3 的进一步改进

模型 3 的改进几乎只是一种形式，但效果进一步显著改善。不仅利用被预测变量的过去值，同时也利用机器人的位置坐标（x，y），便有可能得到一个定位精度极高又非常简洁的模型。

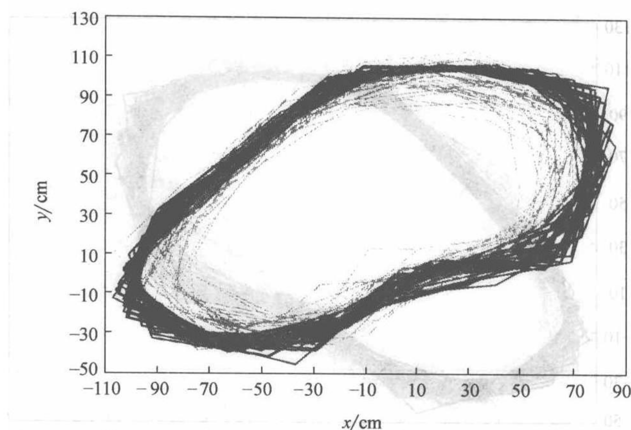


图 6.29　机器人的真实轨迹（粗实线）与表 6.6 和表 6.7 给出的模型 3 由超声波传感器估计出的轨迹（虚线）的比较

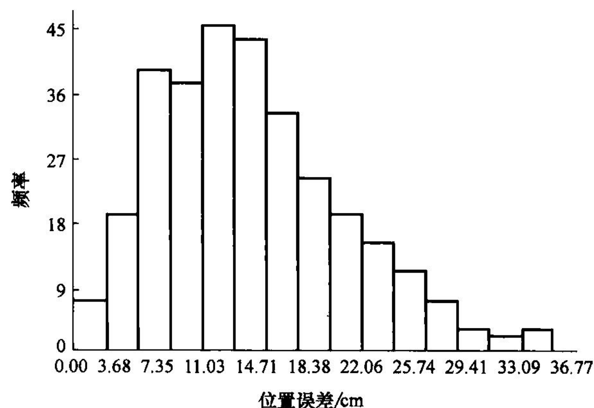


图 6.30　利用表 6.6 和表 6.7 给出了模型 3 中检验数据的定位误差分布

式（6.6）和式（6.7）给出了结果。经过 6s 时间窗的 8 个输入就足以确定机器人的位置，均值定位误差为（10.5 ±0.4）cm。

$$x(t) = 1.04x(t-1) - 0.65y(t-1) - 6.7L_{15}(t) - 8.84L_{30}(t) + 4.05L_{75}(t) + 7.1L_{120}(t) - 5.75L_{135}(t) + 14.53L_{180}(t) \tag{6.6}$$

$$y(t) = -0.49x(t-1) - 0.56y(t-1) - 2.54L_{15}(t) + 11.43L_{30}(t) + 2.19L_{75}(t) + 5.93L_{120}(t) - 2.14L_{135}(t) + 3.10L_{180}(t) \tag{6.7}$$

预测轨迹和真实轨迹如图 6.31 所示。和仅仅依靠以传感器数据为基础的定位比起来，现在的两条轨迹相互吻合得很好。

模型 4 的定位误差分布如图 6.32 所示。

结论

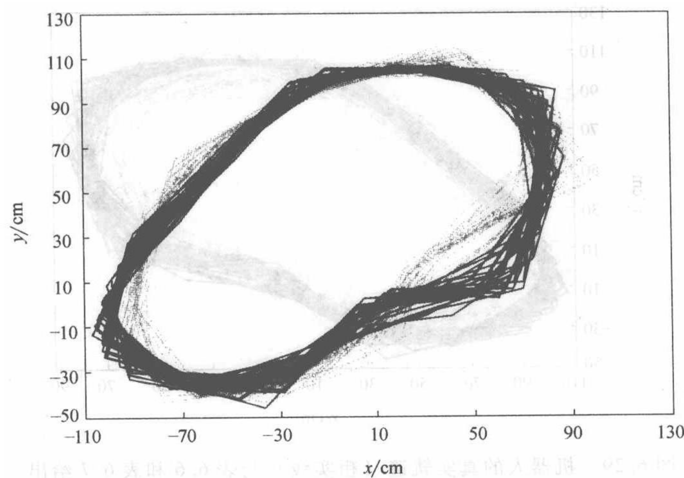


图 6.31 机器人的真实轨迹（粗实线）与表 6.6 和表 6.7 给出的模型 4 中由先前的位置（x，y）以及激光传感器所预测出的轨迹的比较
（实线指的是真实轨迹，虚线是预测的轨迹，均值定位误差是（10.5 ±0.4）cm）

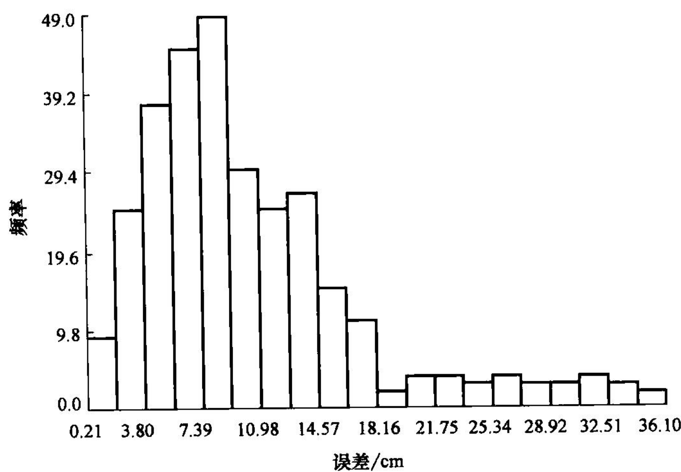


图 6.32 检验数据的定位误差分布
（使用式 6.6 和式 6.7 给出的模型 4）

正如我们在这个例子中所看到的，使用 ARMAX 模型，基于传感器的定位是可实现的。本节讨论了四个降低复杂性但增加精度的模型，表明了机器人的感知向量的哪些部分对自我定位有用，哪些没用。

在第一种模型下，我们证明基于传感器的定位在我们所用的场地上是可以实现的。模型 1 还表明了哪些传感器信息对实验任务特别有用：六个最有用的传感是朝着靠近墙壁的激光传感器。超声波传感器的定位证明不是特别有用。

基于这些考虑我们开发了第二个利用简化的感知输入的模型。在此模型下，我们发现更高的回归阶数对保持同样的定位精度是必须的。也就是说，单独使用传感器信息时，机器人专家可以选择在比较短的时间窗（24s）使用可获得的所有的传感器信息，或者使用在更长时间窗（80s）时较少的传感器信息。

模型3和模型4表明，在该墙壁沿行的例子中，非常准确且简单的模型不仅可以由传感器数据回归而得到，也可以由以前的位置估计得到。这一发现的理由是显而易见的，即墙壁沿行是一种高度可重复和可预见的行为。最终模型达到了机器人直径1/4的均值定位精度，包含仅8个输入量和经过6s时间窗就实现了回归的模型。

6.7 任务辨识

任务辨识的目的是获得机器人控制程序的模型。结果就是把程序代码压缩成一个多项式方程。这样做的直接好处是，在任务代码的实际执行没有什么影响的情况下，使机器人任务的表达更加简捷。

和控制程序类似，任务模型是感知到机器人电动机反馈的映射（见图6.33）。为了获得控制程序的模型，机器人的传感器数据以及对这一传感器数据的反馈在执行控制程序时需要记录下来。这里把传感器数据作为输入、电动机的反馈作为输出，应用用于环境建模的同样的建模技术（见6.6节），以便找到一个控制程序的合适的模型。

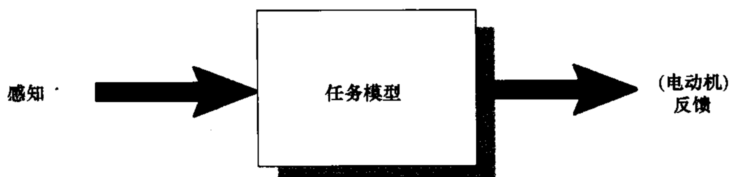


图6.33 任务辨识：一个已知的传感器数据到电动机反馈的函数映射

6.7.1 任务辨识：利用ARMAX辨识墙壁沿行行为

本节介绍了移动机器人任务辨识的一个基本示例，即确定墙壁沿行的一个线性的ARMAX模型。为了使任务辨识的根本机制明确，情况被简单化了：在示例中数据量很小，输入和输出具有很好的相关性，模型获得也是越简单越好。

这个示例的目的是找出移动机器人在一次墙壁沿行运动中传感器数据和电动机反馈的关系。

撇开机器人确切地是如何完成墙壁沿行行为的，我们将看到输入和输出的关系是否能被确立，这里机器人的激光传感器数据向前直行（激光传感器90

号）和右转45°（激光传感器135号）为输入、机器人的拐弯速度（$\dot{\phi}$）为输出。如图6.34所示，该例中所用的实际输入和输出值由表6.8给出，如图6.35所示。

表6.8 图6.35数据的数值（从左往右读）

激光传感器135号(输入)										
	1.64	1.6	1.55	1.54	1.55	1.54	1.62	1.76	1.98	2.04
	1.92	1.89	1.88	1.92	1.91	1.91	1.86	1.79	1.77	1.7
	1.64	1.62	1.57	1.55	1.54	1.53	1.53	1.64	1.71	1.9
	2.01	1.92	1.88	1.87	1.87	1.89	1.89	1.81	1.75	1.71
	1.65	1.63	1.57	1.55	1.54	1.51	1.44	1.57	1.55	1.72
	2.03	1.99	1.93	1.86	1.87	1.92	1.93	1.9	1.78	1.78
	1.7	1.67	1.6	1.58	1.55	1.5	1.48	1.53	1.5	1.49
	1.69	1.94	2.04	1.94	1.89	1.89	1.92	2.02	1.91	1.86
	1.81	1.74	1.71	1.64	1.63	1.6	1.54	1.55	1.57	1.51
	1.64	1.7	1.87	2.01	1.92	1.88	1.89	1.92	1.93	1.87
	1.87	1.78	1.76	1.68	1.67	1.62	1.57	1.55	1.54	1.51
	1.58	1.66	1.67	1.97	2.01	1.91	1.87	1.87	1.9	1.87
	1.89	1.8	1.76	1.69	1.65	1.64	1.57	1.55	1.59	1.48
	1.51	1.52	1.56	1.72	2.1	1.99	1.92	1.87	1.89	1.91
	1.93	1.92	1.81	1.77	1.71	1.67	1.62	1.53	1.57	1.47
	1.49	1.49	1.48	1.48	1.63	1.84	2.06	1.95	1.89	1.89
	1.95	1.99	1.92	1.84	1.81	1.75	1.71	1.65	1.63	1.62
	1.54	1.55	1.52	1.55	1.55	1.75	1.93	2.02	1.93	1.89
	1.91	1.91	1.9	1.92	1.88	1.79	1.75	1.7	1.63	1.66
	1.57	1.58	1.54	1.53	1.53	1.58	1.75	1.8	2.01	1.93
激光传感器90号(输入)										
	1.31	1.24	1.17	1.09	1.04	0.97	0.91	0.89	0.85	0.84
	0.9	1.00	1.12	1.45	1.66	1.76	1.7	1.61	1.52	1.44
	1.37	1.3	1.22	1.16	1.09	1.02	0.96	0.92	0.88	0.85
	0.87	0.91	0.98	1.19	1.37	1.67	1.77	1.7	1.6	1.52
	1.44	1.37	1.29	1.21	1.16	1.08	1.00	0.96	0.91	0.87
	0.86	0.84	0.88	0.97	1.19	1.55	1.8	1.75	1.76	1.59
	1.52	1.43	1.35	1.27	1.23	1.14	1.07	1.00	0.95	0.89
	0.85	0.83	0.82	0.85	0.92	1.14	1.37	1.84	1.77	1.68
	1.61	1.53	1.44	1.38	1.3	1.23	1.16	1.09	1.03	0.96
	0.92	0.87	0.84	0.86	0.89	1.00	1.12	1.47	1.73	1.76
	1.7	1.6	1.51	1.42	1.37	1.29	1.21	1.14	1.11	1.01
	0.97	0.92	0.87	0.86	0.86	0.91	1.01	1.17	1.5	1.54
	1.78	1.68	1.6	1.51	1.45	1.36	1.28	1.19	1.15	1.07
	1.01	0.95	0.9	0.87	0.85	0.84	0.88	0.99	1.2	1.53
	1.82	1.74	1.67	1.58	1.52	1.44	1.35	1.26	1.22	1.14
	1.07	1.00	0.95	0.89	0.85	0.82	0.81	0.85	0.9	1.07
	1.47	1.86	1.77	1.7	1.62	1.54	1.45	1.39	1.31	1.25
	1.17	1.1	1.03	0.97	0.93	0.88	0.86	0.84	0.89	1.00
	1.2	1.36	1.6	1.76	1.7	1.6	1.53	1.44	1.39	1.3
	1.23	1.16	1.1	1.03	0.97	0.91	0.88	0.84	0.87	0.9

（续）

拐弯速度（输出）									
0.05	0.06	0.07	0.07	0.08	0.1	0.13	0.15	0.18	0.21
0.23	0.22	0.2	0.11	0.07	0.01	0.03	0.05	0.03	0.05
0.06	0.06	0.07	0.07	0.08	0.09	0.11	0.12	0.15	0.18
0.21	0.22	0.22	0.17	0.13	0.05	0.	0.03	0.05	0.05
0.06	0.05	0.06	0.07	0.07	0.08	0.11	0.11	0.13	0.16
0.18	0.22	0.23	0.24	0.19	0.09	0.	0.	0.08	0.04
0.06	0.03	0.07	0.06	0.05	0.08	0.08	0.08	0.1	0.15
0.15	0.18	0.22	0.24	0.26	0.21	0.15	0.04	0.03	0.02
0.03	0.06	0.04	0.07	0.05	0.06	0.07	0.07	0.08	0.11
0.12	0.15	0.18	0.21	0.23	0.22	0.2	0.11	0.05	0.04
0.03	0.06	0.03	0.06	0.05	0.06	0.07	0.07	0.07	0.1
0.1	0.13	0.15	0.18	0.21	0.23	0.22	0.18	0.09	0.09
0.01	0.03	0.04	0.05	0.06	0.04	0.07	0.06	0.05	0.09
0.09	0.11	0.14	0.16	0.19	0.22	0.24	0.23	0.18	0.11
0.02	0.04	0.02	0.03	0.06	0.06	0.04	0.08	0.04	0.09
0.09	0.09	0.11	0.1	0.15	0.18	0.22	0.24	0.27	0.23
0.12	0.01	0.02	0.05	0.04	0.05	0.04	0.06	0.06	0.05
0.07	0.07	0.09	0.1	0.12	0.16	0.18	0.21	0.23	0.22
0.17	0.13	0.08	0.	0.02	0.05	0.05	0.05	0.07	0.04
0.07	0.06	0.08	0.09	0.11	0.13	0.15	0.19	0.2	0.22

可以理解的一个最简单的Armax模型是对输出完全没有使用回归的模型（y的回归阶数为零），而对输入有一次回归阶数一个输入的回归次数（如只有当时刻t和$t-1$，输入u可用于估计输出y）。

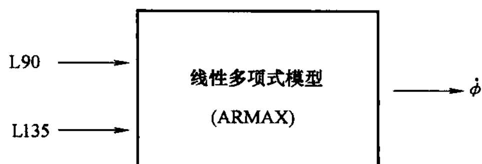


图6.34　任务辨识

在Scilab中，这样的辨识可以通过如下方式得到：

```
armax (0, 1, [y'; zeros (1:200)], u')
A (x)  =
!  1  0  !
!        !
!  0  1  !
B (x)  =
!   -0.0060399   +0.2257918x   -0.1456451   -0.0820220x  !
!                                                         !
!        0                          0                     !
```

建模的结果在式（6.8）中给出：

$$\dot{\phi} = -0.0060399 l_{135}(t) + 0.2257918 l_{135}(t-1) - 0.1456451 l_{90}(t) - 0.0820220 l_{90}(t-1) \quad (6.8)$$

l_{90}和 l_{135}为在方向为 90°（径直行进）和 135°（行进方向偏右 45°）上，机器人的激光测距仪所获得的读数，如图 6.35 所示。

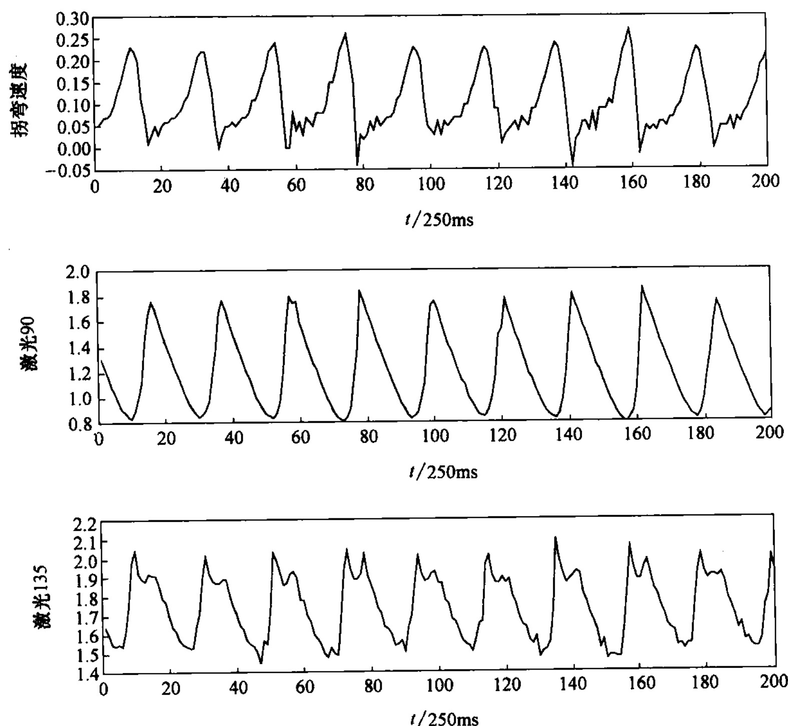


图 6.35 任务辨识例子中使用的输入和输出数据（最上面的图示机器人电动机对传感器数据的反馈（拐弯速度），下面的图是作为模型输入的激光传感器数据（激光传感器 90 号和激光传感器 135 号），三个图的数值在表 6.8 中已给出）

一个简要说明：除了使用上面所用的普通的方法，还可以使用 6.43 节中给出的程序。因为那种程序用前半部分数据获得一个能对后半部分数据进行检验的模型，这个模型与式（6.8）给出的仅有微小的差别，然而与机器人原本的旋转速度非常接近。

如果我们将式（6.8）给出的模型的实际旋转速度 $\dot{\phi}$ 绘成图，我们就可以发现即使这样一个简单的线性模型实际上已经能很好符合（见图 6.36）！

实际上，所有的机器人任务辨识都应该尝试用线性的 Armax 模型；对于建

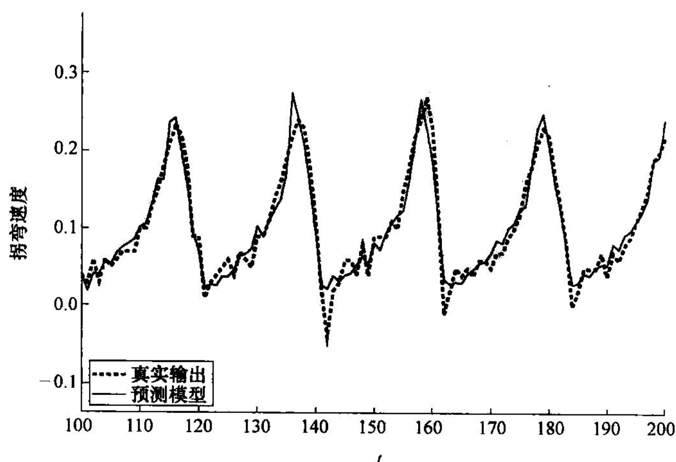


图 6.36　机器人实际的旋转速度（粗虚线）和
式（6.8）给出的多项式模型（细线）的比较

模任务，往往证明它们足够了。对于非线性关系，显然，非线性的 NARMAX 建模是必须的，然而确定它也更为复杂。

6.7.2　任务辨识：用 Narmax 辨识墙壁沿行行为

让我们研究一个相同的问题——在一个更复杂环境下的墙壁沿行。在任务辨识的第二个例子中，一个麦哲伦专业版移动机器人在图 6.37 所示的环境中执行墙壁沿行的任务。

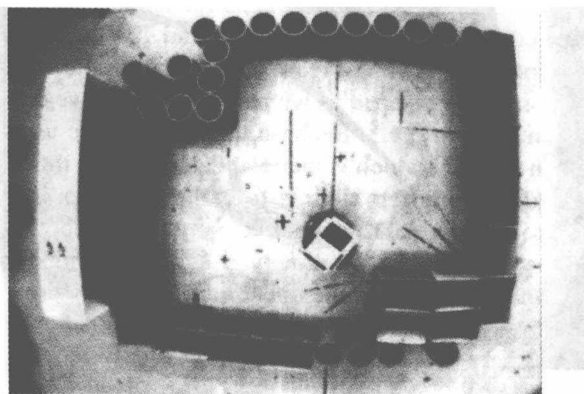


图 6.37　墙壁沿行任务执行的环境

类似于［Iglesias et al, 1998］讨论的方法，通过使用一种已经被训练使用超声波传感器输入来产生正确电动机反馈的返回式传播神经网络，可以实现“初始的”墙壁沿行。由此得到的“初始的”轨迹，如图 6.38 所示。

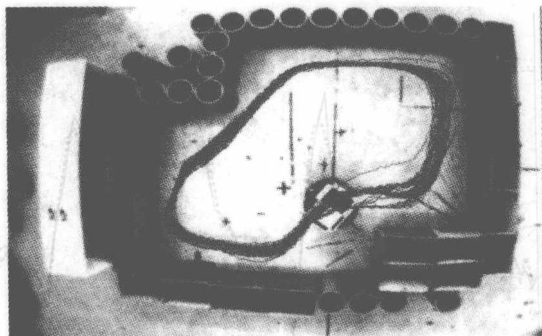


图 6.38 图 6.37 所示的是用人工神经网络控制器得到的图示环境中观察到的墙壁沿行行为

然后我们就可以用 Narmax 方法，辨识这个墙壁沿行任务，得到如表 6.9 所示的模型。

用机器人十六个超声波测距读数，可以得到表 6.9 中给出的模型的 u1 到 u16 16 个输入，并且通过转换，把所有的数值设置在 0.25 到 0[⊖]之间。

下一步，显然是通过 Narmax 的模型运行机器人，而不是初始的神经网络。作为结果的轨迹如图 6.39 所示。

比较图 6.38 和 6.39，可以清楚地看到，两个轨迹图像都非常相似。问题是它们是不是就是同一个呢，我们在下面第 6.9 节将作更进一步的讨论。

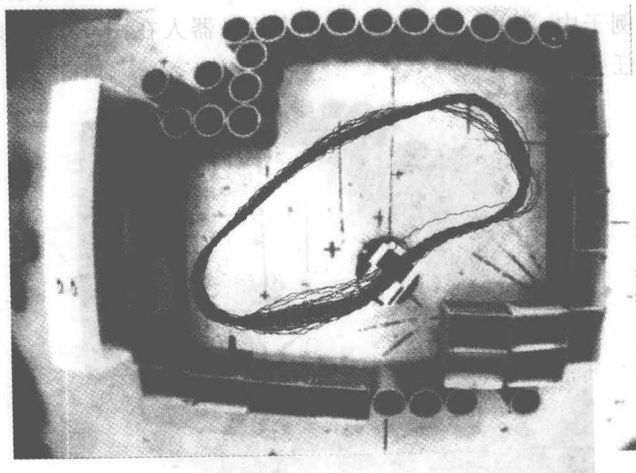


图 6.39 用 Narmax 模型控制的机器人所观察到的轨迹

⊖ 麦哲伦的超声波传感器只返回一个距离达 4m 的有效测距读数。超过这个距离，只返回 4 米。通过把它们设置为零，我们实际上删掉了所有 1/4 的读数。

表 6.9　墙壁沿行的 Narmax 模型（u 指输入 1～16（参见原文说明），n 是建模的旋转速度 r 的时间步长）

$$
\begin{aligned}
r(n) = & -0.3096661 \\
& -0.1243302*u(n,1) \\
& -0.0643841*u(n-2,1) \\
& -0.0389028*u(n,3) \\
& -0.1116723*u(n,9) \\
& +0.1749080*u(n,13) \\
& +0.0897680*u(n,14) \\
& -0.0541738*u(n,15) \\
& -0.0880687*u(n,16) \\
& +0.1128464*u(n,1)^2 \\
& +0.0789251*u(n-2,1)^2 \\
& +0.1859527*u(n,9)^2 \\
& -0.0202462*u(n,13)^2 \\
& +0.0531564*u(n,15)^2 \\
& +0.0996978*u(n,16)^2 \\
& +0.0608442*u(n-1,1)*u(n-1,16) \\
& -0.0507206*u(n-2,1)*u(n-2,9) \\
& +0.0283438*u(n,2)*u(n,14) \\
& +0.0669943*u(n,2)*u(n,16) \\
& -0.0519697*u(n-1,2)*u(n,16) \\
& +0.0714956*u(n,3)*u(n-1,16) \\
& +0.0534592*u(n-1,3)*u(n,15) \\
& -0.0297800*u(n,13)*u(n,14)
\end{aligned}
$$

6.7.3　通过任务辨识的独立平台编程：RobotMODIC 过程

图 6.33 所示的任务辨识情形确定了机器人感知和它的电动机对这一感知的反馈的关系。也就是说，它辨识了机器人的控制程序。这种关系是由一个明白的、可分析的函数来表达的。比如表 6.9 所示的多项式。机器人初始的控制程序可以由机器人工程师通过任何可行的方法来设计，可能是控制论、机器学习技术或者其他途径的；同时，任务辨识使得这一相同任务的表达统一起来。

基于这一考虑的一个显而易见的应用是用任务辨识技术，使一个机器人的行为可以传递给另外一个机器人。这一过程称作 RobotMODIC（机器人建模、辨识和特征表示），如图 6.40 所示。

机器人感知和行为（图 6.40 左边的部分）的最初结合是通过 ARMAX 或者 NARMAX 辨识来实现的（图 6.40 右边的部分）。运用非线性函数可以使电动机相应传感器得到的感知（图 6.40 中间的部分），运用最初的控制器会使机器人显示出可对比的行为。

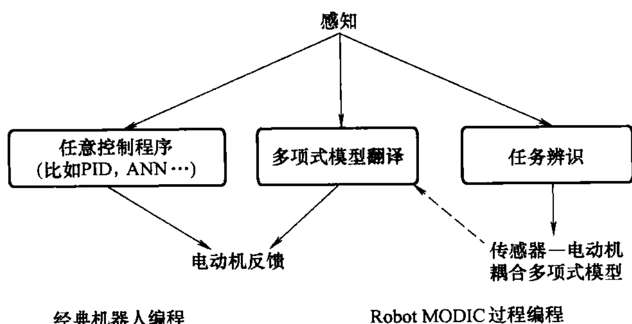


图 6.40 RobotMODIC 过程

机器人辨识这一应用中有趣的一点是，“克隆”的过程既可在原机器人上进行——给原控制器编程提供了可选择的和更简单的方法，又可在另外一个机器人甚至物理上不同的机器人（假设这第二个机器人同原机器人有相似的传感器特性）上进行。实际上，RobotMODIC 过程提供了一个机器人编程的独立平台方案。

实验：机器人 Java

在实际任务辨识工作中进行独立平台的机器人编程的可行性已经被后续的实验验证了，这些实验是我们与圣地亚哥感普特拉大学智能系统研究小组合作进行的。一个进行墙壁沿行的麦哲伦高级机器人的可辨识行为已经在一个 Nomad 200 机器人上实现了（在 Nomad 上生成了一小段控制程序，实质上只由一个多项式组成）。

尽管机器人各有不同，而且麦哲伦与 Nomad 在两个不同的实验室中操作，在圣地亚哥的真实世界环境下，Nomad 可以成功地执行墙壁沿行的行为，在一个多小时的操作中没有任何失误。但这仅仅是一个证明而已，仍不能证明跨平台编程的机器人辨识是可行的。

6.7.4 通过训练的编程：用 RobotMODIC 过程进行门体穿越

通过 RobotMODIC 过程，使机器人控制软件从一个机器人平台到另一个的“翻译”，是一种既便宜又快捷的方法，可以实现许多不同类型机器人相似的机器人行为。可是，它依然有一个弱点：为了辨识机器人的行为并且用线性或非线性的多项式形式表达它，这种行为首先必须在机器人上表现出来。在实践中，这意味着我们不得不先用传统方式写一次机器人控制代码，而目的是为了用多项式的形式重新表达它。

围绕这个费时的过程有这么一种方法：机器人训练。如果我们手动引导机器人完成要求的传感器—电动机任务，并且辨识这种行为，我们就可以避开传统的编程过程，而或多或少地直接得出了这个任务的多项式表达。

下面的实例研究展示了在一个门体穿越的实例中这是如何实现的，传感器一电动机任务非常复杂，要求对机器人运动的精确控制以及在运动的不同阶段运用不同的传感器（激光传感器只能面对前方，而当机器人实际穿越门口时不能使用。这时，不得不使用超声波或红外线传感器）。

图 6.41 展示的是实验装置。在任务辨识后的训练阶段和自动运动阶段，机器人在阴影区域的某处出发到门的左边。门本身宽度相当于两个机器人的直径，这要求对机器人侧面的运动要精确控制，以免卡住。所用的机器人还是图 1.1 所示的麦哲伦专业版 Radix。

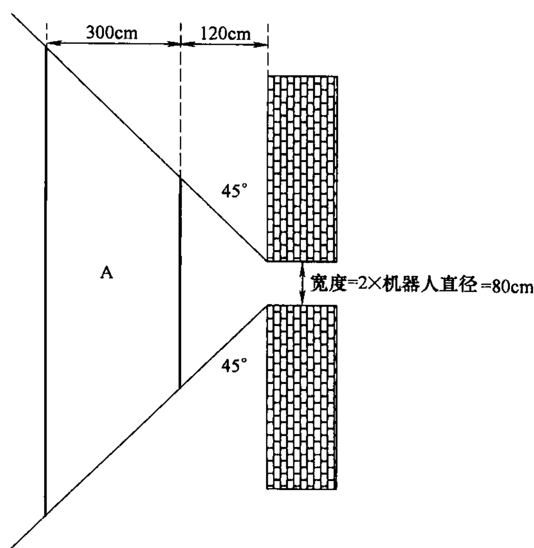


图 6.41　穿越门体行为的实验方案（机器人的起始位置在标为"A"的梯形区域内）

为了得到 RobotMODIC 过程的训练数据，机器人要被放置在图 6.41 所示的 A 区域中的任意起始位置，然后手动操纵穿过门体。在这个操作过程中，所有的传感器和电动机信号都以 4Hz 的采样频率记录下来。收集数据的过程重复了 39 次，产生了 39 组毗邻数据构成一个时间序列，每组由几百个数据点组成。

这些训练数据用来辨识门体穿越的任务，产生了表 6.10 中所示的非线性方程，其中 d 代表 15 度扇形激光测距的平均读数，s 表示超声波测距的读数。

表 6.10　门体穿越行为中角速度 $\dot{\theta}$ 的 NARMAX 模型（激光和超声波信息 d 与 s 的函数）

$$
\begin{aligned}
\dot{\theta}(t) = \; & +0.272 \\
& +0.189 * (1/d_1(t)) \\
& -0.587 * (1/d_3(t)) \\
& -0.088 * (1/d_4(t)) \\
& -0.463 * (1/d_6(t)) \\
& +0.196 * (1/d_8(t)) \\
& +0.113 * (1/d_9(t)) \\
& -1.070 * (1/s_9(t)) \\
& -0.115 * (1/s_{12}(t)) \\
& +0.203 * (1/d_3(t))^2
\end{aligned}
$$

（续）

$-0.260 * (1/d_8(t))^2$
$+0.183 * (1/s_9(t))^2$
$+0.134 * (1/(d_1(t) * d_3(t)))$
$-0.163 * (1/(d_1(t) * d_4(t)))$
$-0.637 * (1/(d_1(t) * d_5(t)))$
$-0.340 * (1/(d_1(t) * d_6(t)))$
$-0.0815 * (1/(d_1(t) * d_8(t)))$
$-0.104 * (1/(d_1(t) * s_8(t)))$
$+0.075 * (1/(d_2(t) * s_7(t)))$
$+0.468 * (1/(d_3(t) * d_5(t)))$
$+0.046 * (1/(d_3(t) * s_5(t)))$
$+0.261 * (1/(d_3(t) * s_{12}))$
$+1.584 * (1/(d_4(t) * d_6(t)))$
$+0.076 * (1/(d_4(t) * s_4(t)))$
$+0.341 * (1/(d_4(t) * s_{12}(t)))$
$-0.837 * (1/(d_5(t) * d_6(t)))$
$+0.360 * (1/(d_5(t) * d_7(t)))$
$-0.787 * (1/(d_6(t) * d_9(t)))$
$+3.145 * (1/(d_6(t) * s_9(t)))$
$-0.084 * (1/(d_6(t) * s_{13}(t)))$
$-0.012 * (1/(d_7(t) * s_{15}(t)))$
$+0.108 * (1/(d_8(t) * s_3(t)))$
$-0.048 * (1/(d_8(t) * s_6(t)))$
$-0.075 * (1/(d_9(t) * d_{16}(t)))$
$-0.105 * (1/(d_{10}(t) * d_{12}(t)))$
$-0.051 * (1/(d_{10}(t) * s_{12}(t)))$
$+0.074 * (1/(d_{11}(t) * s_1(t)))$
$-0.056 * (1/(d_{12}(t) * s_7(t)))$

图 6.42 显示了人类操作者（左图）操作时形成的轨迹，用来生成 RobotMODIC 过程的训练数据，而当机器人在表 6.10 给出的多项式控制下自动运动时产生的轨迹如图 6-42 右图所示。

在两个实例中机器人穿过门口都没有问题。对轨迹的仔细分析表明，与被人所控制相比，自主控制下机器人的运动要平稳得多，并且能够从中央穿过门。

总之，这个对于穿门实例的研究表明，通过机器人训练和后来的任务辨识，得到准确、相对复杂的运动操作是可能的，与用传统方法对机器人编程相比，这个过程要有效得多，也更容易更快速。

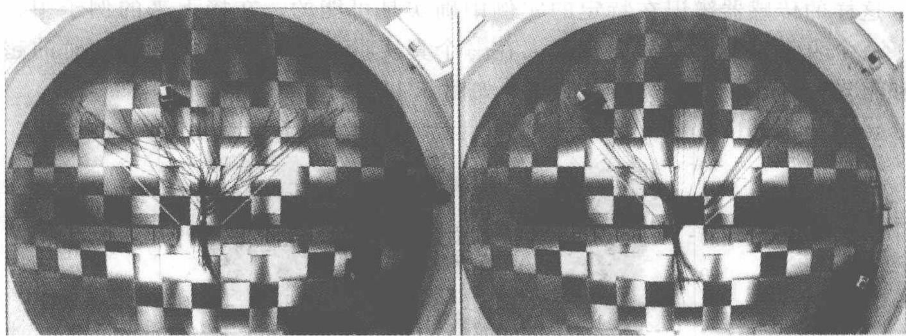


图6.42 左图：在手动操作下的机器人轨迹（39次运动，训练数据）
右图：模型控制下的轨迹（41次运动，检验数据）

6.8 传感器辨识

当然，在6.7.3节所讨论的独立平台的机器人编程需要初始使用的机器人（机器人A）和用经过辨识的函数（多项式）作为控制器的机器人（机器人B）都使用相同类型的传感器。可是，这就有一个问题，如果机器人A使用的是超声波传感器来产生初始的行为，而机器人B只有激光传感器。在这个例子中，所辨识的用来将超声波传感器数据映射到电动机反馈的函数不能用于机器人B。

或者这是可行的？

在特殊的测距传感器中，机器人的许多传感器形式可产生可比较的信号。例如，不是完全相同地，激光测距仪和超声波测距仪都可以得到机器人前方物体相似的深度图。如果我们有从激光到超声波的“翻译”，就可以使用Robot-MODIC过程将机器人A上执行的初始行为转换为机器人B上的等效行为，即使这两个机器人使用的是不同的传感器形式。我们将这种方法称为“传感器辨识”，这个过程如图6.43所示。

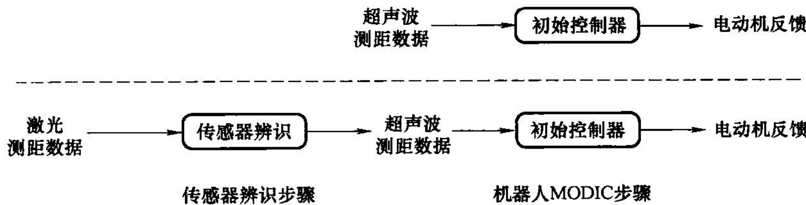


图6.43 用于RobotMODIC过程的传感器辨识（行为产生的初始过程显示在上图。下图描述了使用RobotMODIC过程下的传感器辨识后行为的产生过程）

这样的传感器辨识在特定的实例中确实是可能的。在接下来的例子中，记录了一个麦哲伦专业版移动机器人在一个真实世界的环境中任意移动时的超声波和激光传感器数据。机器人的轨迹如图 6.44 所示。

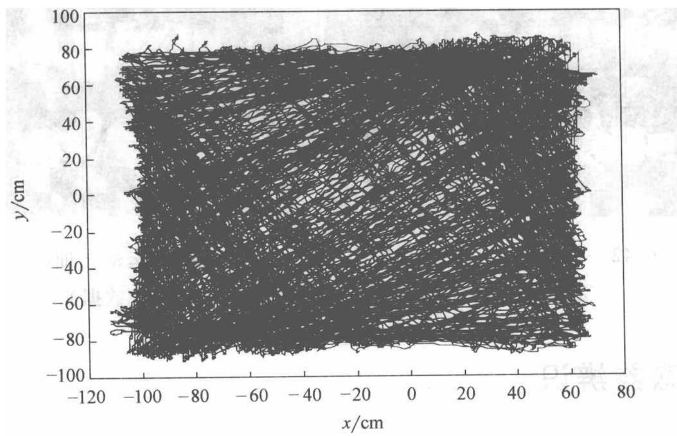


图 6.44 麦哲伦专业版机器人一次任意移动的轨迹（当机器人沿该轨迹运动时，为了接下来的传感器辨识，机器人的超声波和激光传感器数据都被记录下来）

这里我们感兴趣的传感器任务辨识是从激光感知到超声波感知的转换，如图 6.43 所示。为了实现这一过程，我们采用五个激光传感器的测距信号作为输入，而产生的一个超声波传感器信号作为输出，如图 6.45 所示。

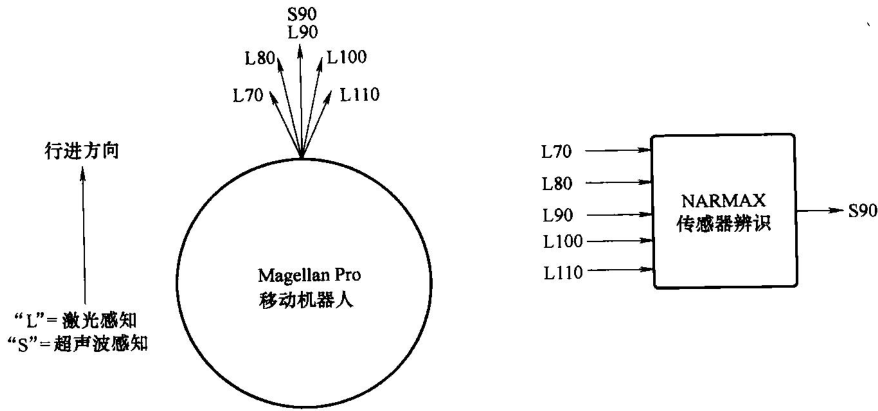


图 6.45 传感器辨识实例中所用的输入与输出数据

NARMAX 模型所得的多项式如表 6.11 所示，而通过五个激光测距的测定，实际超声波传感器数据与预想的超声波传感器数据之间的对应如图 6.46 所示。

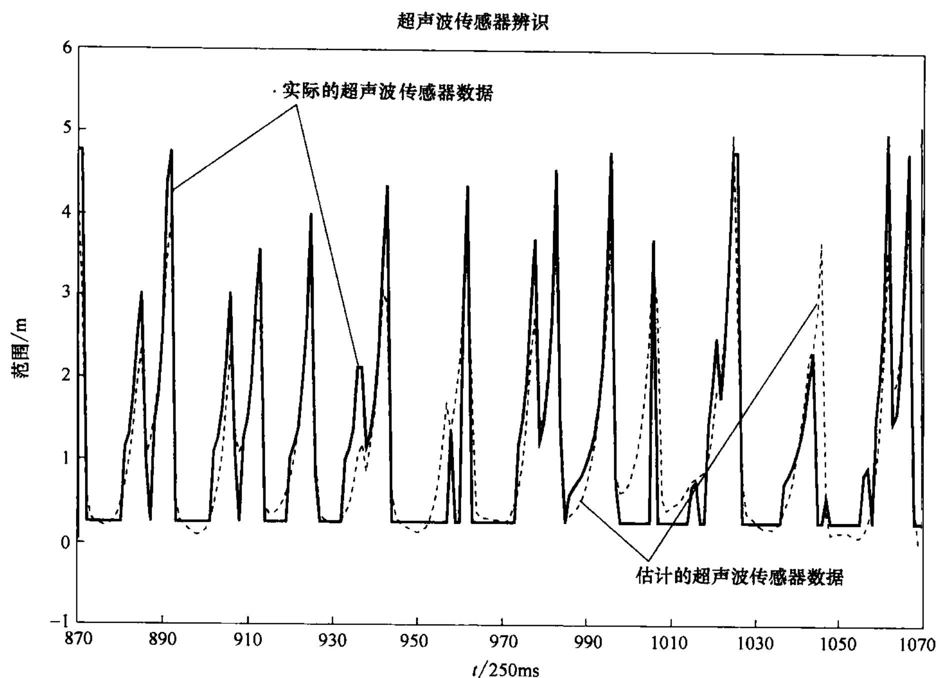


图 6.46　记录下的实际超声波传感器数据（粗实线）与表 6.11 给出模型下估计的超声波传感器数据（细虚线）的对比

表 6.11　激光传感器 L70 ~ L110（五个）到超声波传感器 S90 的数据转换

$$
\begin{aligned}
S90(t) = & -0.09486578751437843571 \\
& +1.34090710655649880678 * L70(t) \\
& -1.46993209143653791315 * L70(t-1) \\
& -0.10133156027639933505 * L70(t-2) \\
& +1.62510983787471263717 * L80(t) \\
& -0.04077783307865779500 * L80(t-2) \\
& -2.58382888545996491914 * L90(t) \\
& +1.34797456078995359086 * L100(t) \\
& -0.30776670071816458751 * L110(t-1) \\
& -1.24584261624820435976 * L70(t)^2 \\
& +0.12259678893997662252 * L70(t-1)^2 \\
& -3.70451454646306554963 * L80(t)^2 \\
& -0.05547179821486561413 * L80(t-1)^2 \\
& -1.54497725705582955591 * L90(t)^2 \\
& +0.08003594836197346074 * L100(t)^2 \\
& -0.11219782488872127868 * L110(t)^2 \\
& -0.22856361726059648554 * L110(t-1)^2
\end{aligned}
$$

（续）

	+1.13094692355659165450 * S90 (t-1)
	-0.10649893253405356974 * S90 $(t-1)^2$
	+1.70065894822170871059 * L70 (t) * L80 (t)
	-0.17397893514679030336 * L70 (t) * L80 (t-2)
	+0.16703072280923750292 * L70 (t) * L90 (t)
	+0.24576961719240705828 * L70 (t) * S90 (t-1)
	+0.51897979560886331463 * L70 (t-1) * L80 (t-2)
	+0.67296902266249047919 * L70 (t-1) * L90 (t)
	-0.21876283482185332474 * L70 (t-1) * u (n-2, 4)
	-0.08831850086310211179 * L70 (t-1) * S90 (t-1)
	+0.21165059817172712786 * L70 (t-2) * L110 (t)
	-0.20194679975663892835 * L70 (t-2) * S90 (t-1)
	+4.51797017772498765709 * L80 (t) * L90 (t)
	-0.42664008070691378238 * L80 (t) * S90 (t-1)
	-0.36439534395116823795 * L80 (t-1) * L100 (t)
	-0.21557583313895936628 * L80 (t-2) * L100 (t-1)
	+0.54408085200311495644 * S90 (t-1) * L110 (t-1)
	+0.30411288763928323586 * L90 (t-2) * L100 (t-1)
	-0.11689726904905589633 * L110 (t-1) * S90 (t-1)

6.9 两种行为何时会相同

6.9.1 行为之间的静态比较

在这一章的前面部分（6.7.2 部分），我们辨识了墙面行走的任务，最初是通过人工神经网络用一个 NARMAX 模型实现的，而后又在机器人上实现这个模型。以原始控制器完成的轨迹如图 6.38 所示，以 NARMAX 模型完成的如图 6.39 所示。两个轨迹共同示于图 6.47 中[⊖]。它们看起来确实很相似，但我们需要遵循科学机器人学原则来量化这种相似之处。

表 6.11 为激光传感器 L70～L110（五个）到超声波传感器 S90 的数据转换。表中符号与图 6.45 的符号对应。

说明图 6.47 所示的两个轨迹的相同与不同之处的一种方法是，可以将两个轨迹实质上当作图像来看。如果在原始像素（机器人的 x 与 y 位置）的分布和模型的行为没有显著差异，至少可以认为有很强的证据证明这两种行为有共同的基本性质。显然这是一个静态的试验，而这种行为很大程度上还与动力学有关。尽管如此，这仍是从统计意义上的初次尝试判断两个轨迹是否相似。

⊖ 两个轨迹的坐标值已经用如下方法归一化了，它们的几何关系被保留下来，方法是将所有的坐标值都除以最大值（x_{ANN}，y_{ANN}，x_{Model}，y_{Model}）。

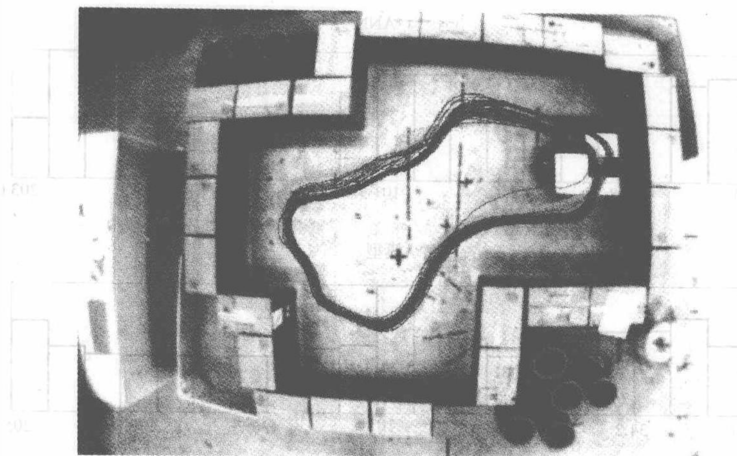


图 6.47　ANN（人工神经网络）墙壁沿行控制器的轨迹（点划线）与 NARMAX 多项式控制器下的轨迹（实线）的对比

机器人原来的运动和模型的 x，y 坐标分布如图 6.48 所示。

根据 3.4.2 节中介绍的曼-惠特尼（Mann-Whitney）U-检验，我们可以发现机器人的两个 x 坐标的分布没有显著的差异（$p<0.05$），但模型的 y 和原来的 y 坐标分布有显著的差异。这也可以从图 6.47 中定性地分析出来。这表明在 x 轴方向上模型化的运动和原来的运动在空间占有上并没有显著差异，但是沿着 y 轴方向，模型驱动的机器人与人工神经网络驱动的机器人在差异分布上却有很大的差别。另外，如果我们测量所有的轨迹使他们占有 [0 1] 的区间，也就是引入一个失真，仅考虑轨迹的几何形状，此时 y 的两个分布的显著差异消失了。

这两个轨迹的比较是静态的对比，因为我们只考虑了机器人在执行它的控制程序时所在位置的空间。然而，机器人和环境之间的相互作用也有重要的动力学方面的性质。这个确实是很重要的，至少在一些特定的应用中，不论是机器人移动得快还是慢，无论它是否会经常地停止等。这些动力学因素不能够通过分析它物理上占有的空间来获得，但是可以根据动力学系统的理论来获得。

6.9.2　行为动力学的比较

在第 4 章里我们讨论了动力学系统的行为，例如一个可以独立移动的机器人既能够通过它所处的物理空间，也可以通过它所处的系统的相空间来描述。后者有可以应用很多工具和定量的测量手段的优势，我们将 4.4.2 节和 4.4.5 节中介绍的方法应用在图 6.47 中的两个轨迹上。

1. 预测时域

图 6.49 表示的是用原人工神经网络控制的机器人在 x 轴方向上移动的预测时域。

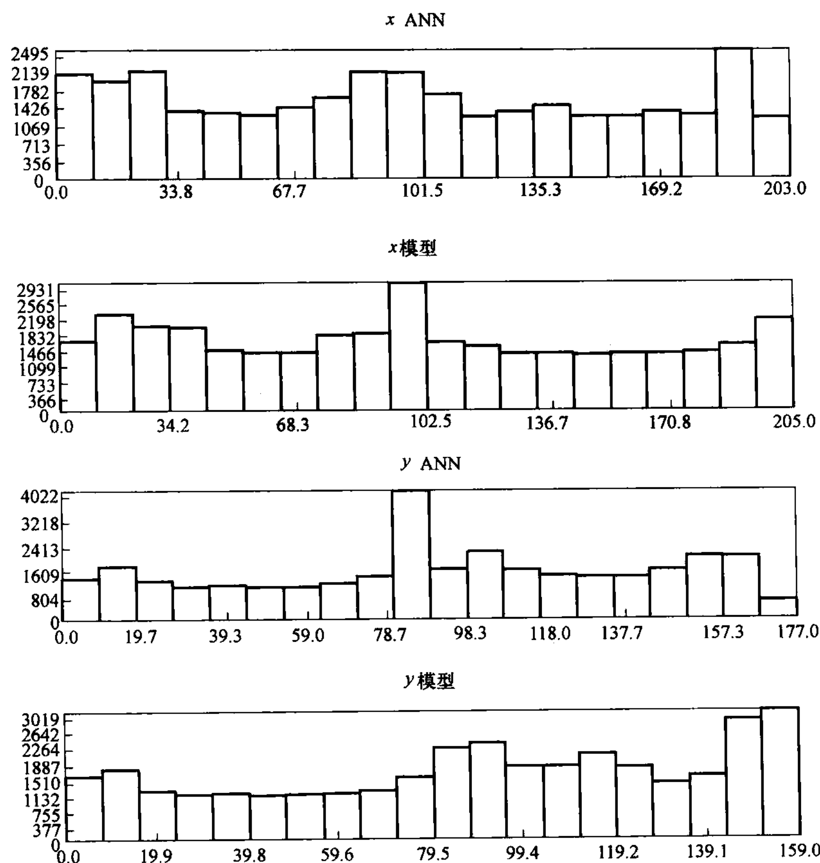


图 6.48　原人工神经网络模型控制的做墙壁沿行运动的机器人和 NARMAX 模型的 x，y 坐标分布的概率

图 6.49 清楚地显示出机器人墙壁沿行的运动非常具有可预测性。当我们试图预测 10000 个时间步（超过 40min）后机器人的 x 坐标时，如果我们用机器人过去的位置来进行预测，比从前一半数据中随机选出的数据作为我们的预测结果，有更小的预测误差，这意味着这个运动从本质上是具有确定性和可预测性。李雅普诺夫指数基本上是零。

图 6.50 表示的是在机器人在执行 NARMAX 模型的墙壁沿行运动时，对机器人 x 坐标做相同的预测。

结果与通过原来的运动获得的结果相似，甚至对于长时间的预测来说，基于数据的预测比基于基线的预测要好。而且，李雅普诺夫指数基本上是零，并且和人工神经网络控制的墙壁沿行运动的机器人的李雅普诺夫指数没有区别。

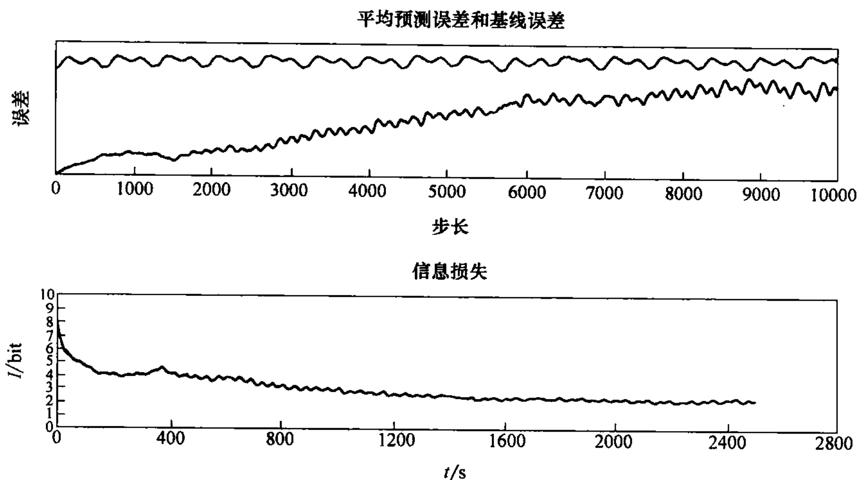


图 6.49　原人工神经网络控制的机器人在 x 轴方向上的移动的预测时域

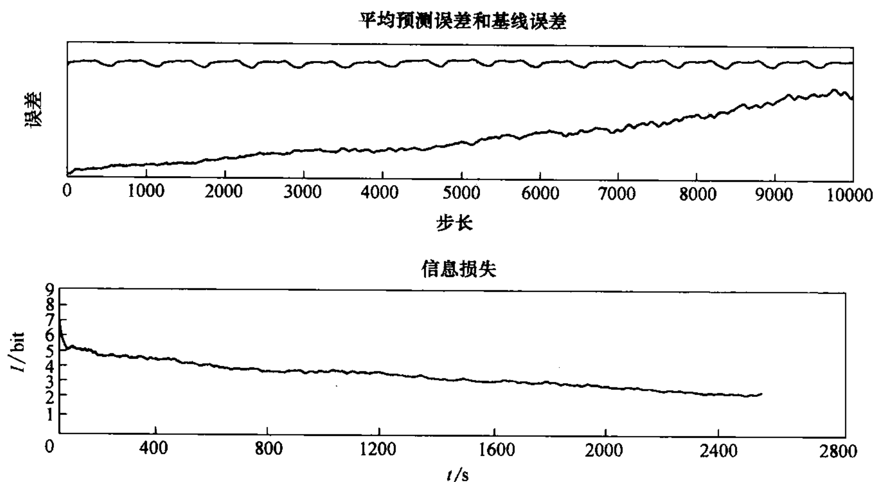


图 6.50　确定的 NARMAX 模型墙壁沿行移动机器人在 x 轴方向上的移动预测时域

2. 使用熵

当综合考虑静力学和动力学的性质的时候，还有另一种量化原运动和模型运动之间的相似却不同的方法：观察原运动和模型运动之间的相互联系。

图 6.51 所示为 1000s 内 x_{ANN}（t）和 x_{Model}（t）的图像，从图像上可以看出这两种情况中机器人在沿着 x 轴运动的图像是相似的，但 x_{Model}（t）更快一些，另外，这两个时间序列有时是同相位的，有时是不同相位的。

这也可从图 6.52 中 x_{ANN}（t）和 x_{Model}（t）的函数图像中得到。

通过分析列联表我们可以得到 x_{ANN}（t）和 x_{Model}（t）之间的相关程度。表

6.12 是 1000 个数据的列联表。

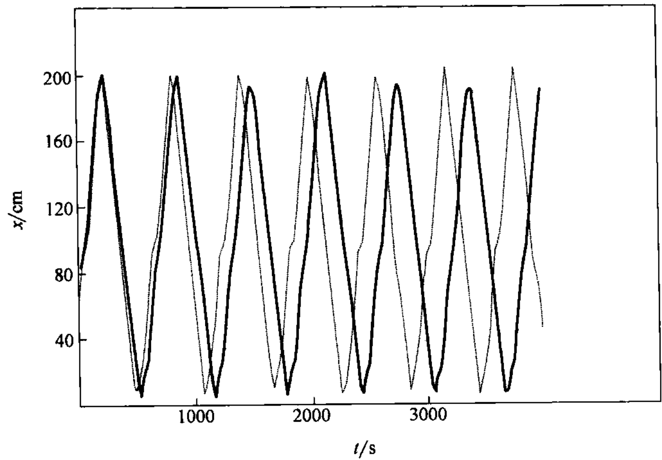


图 6.51　1000s 内原机器人（粗线）和模型机器人（细线）沿着 x 轴方向的运动（可以清晰地看到模型机器人有更高的频率）

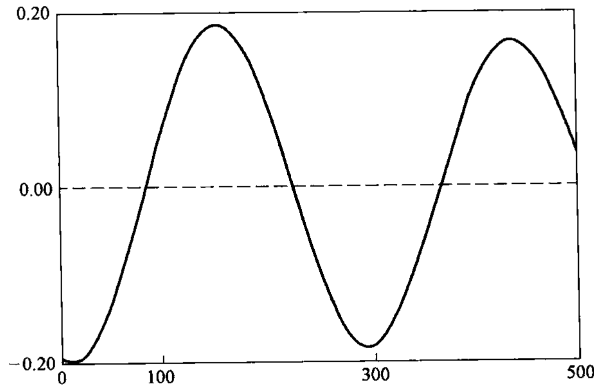


图 6.52　x_{ANN}（t）和 x_{Model}（t）之间的函数图像

表 6.12　x_{orig}（t）和 x_{Model}（t）之间的列联表

57.	84.	61.	47.	172.	83.	58.	42.	24.	1.
51.	29.	26.	32.	104.	8.	14.	18.	23.	29.
58.	35.	12.	41.	61.	24.	4.	8.	16.	48.
57.	26.	61.	37.	37.	26.	33.	17.	31.	83.
88.	88.	32.	29.	60.	33.	33.	72.	78.	43.
57.	45.	13.	16.	36.	39.	48.	29.	21.	37.
74.	0.	22.	19.	33.	59.	49.	20.	20.	41.
49.	24.	1.	48.	48.	29.	25.	37.	33.	30.
29.	23.	36.	57.	40.	18.	30.	38.	54.	27.
5.	28.	60.	55.	29.	43.	37.	37.	43.	75.

从图6.51中可以看到：两条轨迹具有相似的形状，但周期不同。对于前1000个数据来说，这两条轨迹的相位非常相近，但到第3500个数据时，它们轨迹的相位却是相反的。这种情况下，任何列联表的分析都依赖于所考虑的数据数目，事实上不确定性系数将会从1000个数据时的0.30下降到14000个数据时的0.01。对1000到14000个数据进行χ^2分析（参见3.8.1节）时$x_{orig}(t)$和$x_{Model}(t)$之间有显著的相关性，但经过一段很长的时间后，它们的相关性变得很小。这表明尽管原模型和模型有相似的轨迹和相似的李雅普诺夫指数，但在原模型和模型之间存在动力学差异。

6.10　结论

6.10.1　小结

在移动机器人中有一个共同的建模情形是对输入输出之间的关系建模，例如：

1）机器人的位置与感知（仿真）。

2）感知与机器人的位置（自我定位）。

3）感知与电机反馈（机器人编程）。

4）感知与感知（传感器信号的翻译）。

本章证明了这些模型是可以通过应用诸如人工神经网络（6.3节）的方法获得的。这种方法的缺点是，尤其是在考虑到从科学移动机器人到分析机器人行为的目的时，人工神经网络模型是含糊的，因此尽管输入与输出之间的关系是被模型化了的，但有很大程度上是不清楚的。

因此，我们引进一种能够通过应用多项式（ARMAX和非线性的ARMAX模型）进行数学分析的可供选择的明了的方法。

辨识输入输出关系大体上通常指系统检验，相应地，机器人的应用指机器人辨识。本章中给出的例子表明通过机器人辨识，建立机器人和环境之间交互作用的可靠仿真器是可能的（6.6节），机器人的自定位是可能的（6.6.3节），跨平台编程（机器人Java）是可能的（6.7节），信号能够从一个传感器形式“翻译”到另一个传感器形式（6.8节）。这些技术为机器人的发展、分析、机器人控制编码的改进提供了一套强有力的工具。

6.10.2　开放性问题

在所有的建模工作中，最大的问题是获得的模型对于原来的机器人是否可靠，也就是，在进行抽象和概括后，模型是否还保持着原来机器人的基本性质。

这是一个很难回答的问题，而且很大程度上依赖于可靠的定义。在 6.9 节中我们给出了一些比较两个机器人的运动的例子：①比较机器人的静态轨迹；②比较两种情况下机器人运动的动力学信息。当然这些并不是比较机器人运动的唯一方式，其他比较机器人运动的方式也是可以的。

如果两个输出流——初始的和它的模型——进行比较，它们之间的相互关系就能够被确定，并能够分析其显著性水平。在本章中，这是通过相关系数，如斯皮尔曼（Spearman）等级相关系数 r_S 对传感器进行建模的。另外，可以用列联表分析（x^2，克莱默的 V 和基于熵的测量值）来确定原机器人和模型之间相关性是否显著。

第7章 结　　论

概要：本章总结了这本书里提到的材料，并且得出了一些结论。同时也指出在移动机器人研究方面的开放性问题和尚待解决的问题。

7.1 动机

在科学和工业中，机器人学变得越来越重要。一方面归因于拓宽的工业应用链，从现今的普通运输、检查、监视和保洁工作，到有害物质处理的商业应用，或在混乱环境下工作，比如在灾后的废墟中。另一方面，移动机器人作为行为科学中研究行为基础和用来更好了解机器人、任务和环境之间的关系的工具——换句话说，这三个方面中哪一个参数产生哪一种行为，正在变得更有影响力。本书主要关心的是后者。为此，本书里讨论这一内容的第一个原因是有关下面这个问题的：

一方面机器人硬件、机器人软件和环境的关系是什么，另一方面，机器人行为由此产生什么结果？这一点有时称为“机器人与环境交互理论”。

写这本书还有第二个动机。由于至今仍缺乏科学工具与描述行为的方法，大多机器人学研究还局限在存在性证据的陈述。不像其他的自然科学，移动机器人学研究一般不使用对结果的独立重复和证实。这主要是因为我们还没找到相应的“语言”，以一种能够简单重复与证实的实现方式来表达我们的结果。

一种超出现有存在性证据的研究实践——说明某些事情可以做的结论的一次性表述，而不是某些事情一般来说如何去做——它用到了对实验的独立重复与验证，要求可以对结果进行量化的、可测量的描述。因此，本书的第二个动机可以表达为下面这个问题：

机器人—环境的交互怎样量化地描述。如机器人—环境的交互如何被测量，可以使结果是可重复的和可被验证的？

对这两个问题的满意答案会使机器人学迈向加强机器人研究实践的新阶段。我们不仅会理解“如何?”，也会理解“一般而言如何?”。如果我们能证明控制机器人—环境交互的基本理论，然后应用这些理论设计基于此理论理解的任务实现型机器人，而不是用试错法。我们将可以基于对此理论的理解，来预测机器人的行为，并对机器人操作的安全性和稳定性进行分析。

移动机器人学将会是一门主要基于真实机器人实验的学科。即使最好的模

型也替代不了真实的东西，但是通过机器人—环境交互理论的理解和量化描述，实验会变得更清晰和有效。

7.2 机器人—环境交互的量化描述

移动机器人—环境的交互主要有两个方面，它们都是我们要考察的：静态的方面，比如对比两个在空间上而不是在时间上的行为；动态的方面，它体现为机器人的行为是空间和时间的函数。

7.2.1 机器人—环境交互的静态量化分析

在移动机器人研究中的一种典型的方案就是比较同一问题的两种或多种解决方法。例如，在机器人自我定位过程中，我们也许会对基于地标的定位是否比基于航位推测法的定位的表现更好感兴趣。在这个例子中，“更好”是什么意思呢？假设两个系统的表现可以以某种方式记录下来——对于定位方案来说，建立一些把真实位置和感知位置联系起来的列联表（3.8.1 节）是一种很合适的方法——统计方法给我们许多工具可以对性能进行量化的比较，并确定它们之间是否在统计上有显著差异。

第 3 章提供了多种方法，这些方法广泛应用于生物学及心理学中，也可适用于移动机器人学。这些方法确定两种分布是否不同（3.3 节中正态分布的数据，3.4 节中任意分布的数据），序列是否是随机的（3.5 节），两组数值数据是否有相关性（3.6 节和 3.7 节），或两组分类数据之间是否有相关性（3.8 节）。

所有这些检验都是对比检验的，检验中，机器人在一种情况下的行为（一种机器人—任务—环境三者关系的特定表现），与机器人、任务和环境中的一个因素改变的情况进行比较（例如通过改变机器人的控制代码，即任务）。这些检验不考虑机器人—环境交互的动力学。

7.2.2 机器人—环境交互的动力学量化分析

我们希望有第二种量化描述来描述机器人行为，即得到它的行为与环境交互的动力学描述。达到这一目标的一种方法是用动力系统理论，在第 4 章已经讨论过。

一个移动机器人，与它所在的环境交互，是一个动力系统，也就是说是一个描述行为时必须考虑时间作用的系统。换句话说，一个移动机器人与环境的交互可以由微分方程来描述。

在第 4 章里讨论的工具最初是描述行为由微分或差分方程控制的物理系统，现在被称为确定性混沌。这个术语指以下的事实：这些系统不是随机的，而且

仅仅是部分可预测和表现的。在某些情况下，好像它们就是随机的。

确定性混沌理论仅仅适用于确定的和平稳的信号。首先已经建立了那两个事实，然后机器人—环境交互的动力学分析典型地通过时滞嵌入以重构机器人的相空间开始。

一旦相空间被重构，那么它就可以被量化地描述，例如通过估计李雅普诺夫指数（4.4节）、吸引子相关维数（4.5节）或超过它，系统就变得不可预测的预测时域（4.4.2节）。

7.3 机器人—环境交互理论

正如本书中所讨论的，移动机器人学研究的最终目标，是建立一个假设的、概念上的、实用的概括和原理的统一体。这些原理形成了进行移动机器人学研究的普遍参考框架——“机器人—环境交互理论”。

这种理论将产生两个非常有用的好处：

1）这个理论允许检验假设的建立，在“常态科学”的进行中，这是一个必不可少的部分 [Kuhn，1964]。

2）这个理论可以做出预测（例如关于实验结果），因此可以作为对未建立的和没有强大证据支持的假设的证明。

换句话说，这个理论包括了抽象和概括机器人—任务—环境三者间的基本情况。这种概括是必须的：它加强了机器人—环境交互的重要方面，同时弱化了不重要的。最后，一个理论的正确性（或者相反）能通过评价应用这个理论做出的预测而确定。

这样的一个理论会采用什么样的形式呢？显然，它可以用描述各种机器人行为变量间关系的数学描述的形式（公式）来表示，像力、质量和加速度之间的关系（牛顿定律）一样。如果一些变量值已知，那么剩下的变量是可以预测的。

因为一个理论应该形成可检验和可证伪的假设 [Popper，1959]，所以有另一种建立理论的方法：不是试图阐明复杂的机器人与环境之间的交互关系和尝试用数学公式来表达它的所有方面，而是建立机器人与环境交互的模型，并且用这些来形成可检验的假设。这一方法在第6章讨论过。

如果一个计算机模型要获得机器人行为的特性，它必须从真实数据中形成，而不是从有可能基于一个还不存在的理论的考虑。在第6章，我们提出了一个这样的理论。基于已经建立的系统辨识理论 [Eykhoff，1974，Ljung，1987]，我们用所谓的“机器人辨识”过程，建立控制机器人行为的变量间的数学关系。这种过程的好处是多方面的：

1）机器人辨识可以通过能用已建立的理论分析和明确的函数表达式表达关系。

2）机器人辨识允许控制代码在多个机器人平台间的转换，无须为不同的机器人重写代码。

3）它允许对控制代码只做简单的修改使一种传感器形式代替另一种，而无须重写全部的控制代码。

4）它允许可靠的机器人仿真器的建立，支持控制代码的离线编程。

5）它允许仿真器上两个不同的控制机构的精确比较，因为机器人—环境交互的基本模型保持不变。

6）机器人辨识作出关于机器人行为可检验的预测。

7.4 前景：走向分析机器人学

移动机器人学是一门实践性非常强的学科，它涉及到制造能在真实世界中执行真实任务的机器。无论这些任务是“工厂型”任务，如运输和清扫，还是“科研型”任务，如学习和自动导航，在讨论中都无关紧要；在任何情况下，感知和行动的耦合都必须基于设计者提供的机器人控制代码。像在所有的工程任务中，设计得益于对分析的理解。这也是写这本书的一个目的之一，即把这种分析加入当前机器人学研究的议事日程。

与实验移动机器人学相比（目前实验移动机器人学是机器人学研究的主导方面），科学移动机器人学有如下特点（2.8 节）：

1）实验设计和过程是受可检验、可证伪的假设指导的，而不是基于研究者的个人经验（如基于“直觉”）。

2）实验设计和过程是“问题驱动”的，而不是“应用驱动”的。

3）结果是可量化测量和分析的，而不是定性的。

4）实验结果是可独立重复和验证的（例如通过其他的研究小组），而不是作为孤立的存在性证据提出的。

这种研究动力系统实验的方法——不仅局限于移动机器人，而且与动物或技术系统的行为的相关性——依赖于下面三个支柱。

1）描述。

2）建模。

3）分析。

科学的上下文中的描述，一定意味着对实验结果的量化描述，如允许对不同实验观察值进行精确比较的可测量的行为指示器。像第 3 章中所讨论的统计学理论和第 4 章里所讨论的行为动力学的描述子，能达到这个目的。

建模是第二个支柱，因为一个能获得抽象的建模行为特性的模型，可以提供一个集中和凝练的对那些紧要方面的陈述，省略那些无关紧要的。它简化了对所研究系统的理解。第6章的讨论就朝这个目标迈进了一步。

一旦得到一个所研究系统的模型，我们就想知道：

1）这个模型准确吗？比如对建模的系统来说可靠吗？

2）这个模型意味着什么？比如，我们怎样解释这个模型，我们从这个模型里能了解到关于原系统行为的什么东西？

3）这个模型带来什么新的领悟吗？

分析包含了原系统行为和它的模型间的对比。在6.9节提出了一些静态和动态的技术来处理这些问题，但是这里有更多的工作要做，来阐明我们所谓的“同样的”行为准确是指什么。

一些对于模型的理解可以实现，例如通过敏感性分析。最近的在艾塞克斯进行的工作表明，像［Sobol，1993］中讨论的方法能用于量化特定的模型项的重要性，区分模型成分的主次，真的可以带来新的理解。例如，只有通过建立机器人门体穿越行为的模型，然后通过敏感性分析来分析模型后，我们才意识到事实上机器人通过门时，只用到了右边的传感器——对于像通过一扇门这样的“对称”任务而言的，这是一个完全意想不到的结果。理论理解的好处——在这种通过机器人运行的计算机模型表达的情况下——通过下面的例子说明：已经知道一个任务（例如门体穿越）仅仅需要确定的传感器。机器人的硬件和软件可以简化，带来进行更少计算的成本更低的机器人。

描述、建模和分析，是这本书里研究的三个支柱，本书尝试界定一个移动机器人学研究的新日程。然而我们才刚刚开始，并且本书中给出的例子和案例研究只是解决上述问题的一种可能途径。将来的工作必须加深如下问题的处理，如行为的特性、模型的意义、物理智能体操作和它们的仿真间的关系，及计算机建模的理论限制（见［Oreskes et al，1994］关于这个特定问题的讨论）。

因此，这本书是一个对探讨和进一步发展的探索，使我们进一步改进和集中我们的研究和实验到这个新兴的但令人激动的分析机器人学新领域。

参考文献

[Abarbanel, 1996] Abarbanel, H. (1996). Analysis of observed chaotic data. Springer Verlag, New York.

[ANS, 2003] ANS (2003). Tools for dynamics. Applied Nonlinear Sciences, http://www.zweb.com/apnonlin.

[Arkin, 1998] Arkin, R. (1998). Behavior-based robotics. MIT Press, Cambridge, Mass.

[Arkin and Hobbs, 1992] Arkin, R. C. and Hobbs, J. (1992). Dimensions of communication and social organization in multi-agent robotics systems. In From animals to animats 2, Cambridge MA. MIT Press.

[Bacon, 1878] Bacon, F. (1878). Novum Organum. Clarendon Press, Oxford.

[Baker and Gollub, 1996] Baker, G. and Gollub, J. (1996). Chaotic Dynamics. Cambridge University Press, Cambridge, UK.

[Barnard et al, 1993] Barnard, C., Gilbert, F., and McGregor, P. (1993). Asking questions in biology. Longman Scientific and Technical, Harlow, UK.

[Bendat and Piesol, 2000] Bendat, J. S. and Piesol, A. G. (2000). Random data: analysis and measurement procedures. Wiley, New York.

[Beni and Wang, 1989] Beni, G. and Wang, J. (1989). Swarm intelligence in cellular robotic systems. In Nato advanced workshop on robotics and biological systems, Ⅱ Ciocco, Italy.

[Box et al, 1994] Box, G., Jenkins, G., and Reinsel, G. (1994). Time Series Analysis. Prentice-Hall.

[Braitenberg, 1987] Braitenberg, V. (1987). Vehicles. MIT Press, Cambridge, Mass.

[Burgard et al, 1998] Burgard, W., Cremers, A. B., Fox, D., Hähnel), D., Lakemeyer, G., Schulz, D., Steiner, W., Thrun, S. (1998). Experiences with an interactive museum tour-guide robot. Artificial Intelligence, 114: 3-55.

[chen and Billings, 1989] chen, S. and Billings, S. A. (1989). Representations of non-linear systems: The narmax model. Int. J. Control, 49: 1013-1032.

[Critchlow, 1985] Critchlow, A. (1985). Introduction to Robotics. Macmillan, New York.

[Demiris and Birk, 2000] Demiris, J. and Birk, A., (Eds.) (2000). Inter- disciplinary Approaches to Robot Learning. World Scientific Pulishing.

[Dorigo and Colombetti, 1997] Dorigo, M. and Colombetti, M. (1997). Robot Shaping: An Experiment in Behavior Engineering. MIT Press.

[Dudek and Jenkin, 2000] Dudek, G. and Jenkin, M. (2000). Computational principles of mobile robotics. Cambridge University Press, Cambridge, UK.

[Dunbar, 2003] Dunbar, K. (2003). Scientific thought. In Nadel, L., (Ed.), Encyclopedia

of Cognitive Science, London. Nature Publishing Group.

[EXN, 2003] EXN (2003). Discovery channel. http://www.exn.ca/Stories/2003/-03/11/57.asp.

[Eykhoff, 1974] Eykhoff, P. (1974). System identification: parameter and state estimation. Wiley, New York.

[Franklin, 1996] Franklin, J., (Ed.) (1996). Recent Advances in Robot Learning. Kluwer Academic Pulishers.

[Fraser and Swinney, 1986] Fraser, A. M. and Swinney, H. L. (1986). A, 33: 1134-1140. Independent coordinates for strange attactors from mutual information. Physical Review A, 33: 1134-1140.

[Fuller, 1999] Fuller, J. L. (1999). Robotics: introduction, programming, and projects. Prentice-Hall.

[Gillies, 1996] Gillies, D. (1996). Artificial intelligence and scientific method. Oxford University Press.

[Gower, 1997] Gower, B. (1997). Scientific method: a historical and philosophical introduction. Routledge, London.

[Harris, 1970] Harris, E. (1970). Hypothesis and perception. George Allen and Unwin Ltd.

[Iagnemma and Dubowsky, 2004] Iagnemma, K. and Dubowsky, S. (2004). Mobile Robots in Rough Terrain: Estimation, Motion Planning, and Control with Application to Planetary Rovers. Springer Verlag.

[Iglesias et al, 1998] Iglesias, R., Regueiro, C. V., Correa, J., Schez, E., and Barro, S. (1998). Improving wall following behavior in a mobile robot using reinforcement learning. In proc. Of the International ICSC Symposium on Engineering of Intelligent Systems. ICSC Academic Press.

[Kantz and Schreiber, 1997] Kantz, H. and Schreiber, T. (1997). Nonlinear time series analysis. Cambridge University Press, Cambridge.

[Kantz and Schreiber, 2003] Kantz, H. and Schreiber, T. (2003). Tisean - Nonlinear time series analysis. http://www.mpipkd-dresden.mpg.de/tisean.

[Kaplan and Glass, 1995] Kaplan, D. and Glass, D. (1995). Understanding nonlinear dynamics. Springer Verlag, New York.

[Katevas, 2001] Katevas, N., (Ed.) (2001). Mobile Robotics in Health Care Services. IOS Press.

[Kennel and Isabelle, 1992] Kennel, M. and Isabelle, S. (1992). Method to distinguish possible chaos from colored noise and to determine embedding parameters. Phys. Rev. A, 46: 3111-3118.

[Kennel et al, 1992] Kennel, M. B., Brown, R., and Abarbanel, H. D. I. (1992). Determining embedding dimension for phase-space reconstruction using a geometrical construction. Physical Review A, 45: 3403-3411.

[Klayman and Ha, 1987] Klayman, J. and Ha, Y. (1987). Confirmation, disconfirmation and

information in hypothesis testing. Psychological Review, 94: 211-228.

[Korenberg et al, 1988] Korenberg, M., Billings, S., Liu, Y., and McIlroy, P. (1988). Orthogonal parameter estimation for non-linear stochastic system. Int. J. Control, 48: 193-210.

[Kube and Zhang, 1992] Kube, C. and Zhang, H. (1992). Collective robotic intelligence. In Meyer, J., Roitblat, H., and Wilson, S., (Eds.), From Animals to Animats 2, Cambridge MA. MIT Press.

[Kuhn, 1964] Kuhn, T. (1964). The structure of scientific revolutions. University of Chicago Press, Chicago.

[Kurz, 1994] Kurz, A. (1994). Lernende steuerung eines autonomen mobile Roboters. VDI Verlag, Düsseldorf.

[Levin and Rubin, 1980] Levin, R. and Rubin, D. (1980). Applied elementary statistics. Prentice-Hall, Englewood Cliffs.

[Ljung, 1987] Ljung, L. (1987). System identification: theory for the user. Prentice-Hall.

[Loeb, 1918] Loeb, J. (1918). Forced Movements, tropisms and animal conduct. J. B. Lippencott, Philadelphia.

[Mane, 1981] Mane, R. (1981). On the dimension of the compact invariant set of certain nonlinear maps. In lecture Notes in Mathematics 898, pages 230-242, Berlin, Heidelberg, New York. Springer Verlag.

[Martin, 2001] Martin, F. G. (2001). Robotic explorations, a hands-on introduction to engineering. Prentice Hall.

[Mataric, 1994] Mataric, M. (1994). Learing to behave socially. In Cliff, D., Husbands, P., Meyer, J., and Wilson, S., (Eds.), From Animals to Animats 3, Cambridge MA. MIT Press.

[McKerrow, 1991] McKerrow, P. (1991). Introduction to Robotics. Addison-Wesley, Sydney.

[Merriam Webster, 2005] Merriam Webster (2005). Online dictionary. http://www.m-w.com/.

[Morik, 1999] Morik, K., (Ed.) (1999). Making Robots Smarter: Combining Sensing and Action Through Robot Learning. Kluwer Academic Pulishers.

[Murphy, 2000] Murphy, R. (2000). Introduction to AI Robotics. MIT Press, Cambridge, Mass.

[Nehmzow, 2003a] Nehmzow, U. (2003a). Mobile Robotics: A Practical Introduction. Springer, Berlin, Heidelberg, London, New York.

[Nehmzow, 2003b] Nehmzow, U. (2003b). Navigation. Encyclopedia of Cognitive Science.

[Nola and Sankey, 2000] Nola, R. and Sankey, H. (2000). A selective survey of theories of scientific method. In Nola, R. and Sankey, H., (Eds.), After Popper, Kuhn and Feyerabend, Dordrecht. Kluwer.

[Oreskes et al, 1994] Oreskes, N., Shrader-Frechette, K., and Belitz, K. (1994). Verification, validation and confirmation of numerical models in the earth sciences. Science, 263.

[Parker, 1994] Parker, L. (1994). Heterogeneous multi-robot cooperation. PhD thesis, Massachussetts Institute of Technology, Department of Electrical Engineering and Computer Science.

[Paul and Elder, 2004] Paul, R. and Elder, L. (2004). The miniature guide to critical thinking. Foundation for Critical Thinking, Dillon Beach CA.

[Pearson, 1999] Pearson, R. (1999). Discrete-time dynamic models. Oxford University Press, Oxford.

[Peitgen et al, 1992] Peitgen, H., Jürgens, H., and Saupe, D. (1992). Chaos and fractals - new frontiers of science. Springer Verlag, New York, Berlin, Heidelberg, London.

[Pena et al, 2001] Pena, D., Tiao, G., and Tsay, R., (Eds.) (2001). A course in time series analysis. Wiley, New York.

[Popper, 1959] Popper, K. (1959). The logic of scientific discovery. Hutchinson, London.

[Popper, 1963] Popper, K. (1963). Conjectures and refutations: the growth of scientific knowledge. Routledge and K. Paul, London.

[Popper, 1972] Popper, K. (1972). Objective knowledge. Clarendon Press, Oxford.

[Ritter et al, 2000] Ritter, H., Cruse, H., and Dean, J., (Eds.) (2000). Prerationnal intelligence, Dordrecht. Kluwer.

[Rosenstein et al, 1994] Rosenstein, M. T., Collins, J. J., and De Luca, C. J. (1994). Reconstruction expansion as a geometry-based framework for choosing proper delay times. Physica D, 73: 82-98.

[Sacha, 1982] Sachs, L. (1982). Applied statistics. Springer Verlag, Berlin, Heidelberg, New York.

[Schöner and Kelso, 1988] Schöner, G. and Kelso, J. (1988). Dynamic pattern generation in behavioral and neural systems. Science, 239: 1513-1520.

[Scilab Consortium, 2004] Scilab Consortium (1989-2004). The Scilab programming language. http://www.scilab.org.

[Siegwart and Nourbakhsh, 2004] Siegwart, R. and Nourbakhsh, I. R. (2004). Introduction to Autonomous Mobile Robots. MIT Press.

[Smithers, 1995] Smithers, T. (1995). On quantitative performance measures of robot behavior. Robotics and Autonomous Systems, 15 (1-2): 107-133.

[Sobol, 1993] Sobol, I. (1993). Sensitivity estimates for nonlinear and mathematical models. Mathematical Modelling and Computational Experiment, 1: 407-414.

[Steels, 1995] Steels, L., (Ed.) (1995). The biology and technology of intelligent autonomous agens, Berlin, Heidelberg, New York, Springer Verlag.

[Takens, 1981] Takens, F. (1981). Detecting strange attractors in turbulence. In Dynamical Systems and Turbulence, pages 366-381, Berlin, Heidelberg, New York. Springer Verlag.

[Theiler and Lookman, 1993] Theiler, J. and Lookmann, T. (1993). Statistical error in a chord estimator of the correlation dimension: the rule of 'five'. Bifurcation and Chaos, 3: 765-771.

[Ueyama et al, 1992] Ueyama, T., Fukuda, T., and Arai, F. (1992). Configuration and communication structure for distributed intelligent robot system. IEEE Trans. Robotics and Automation, pages 807-812.

[von Randow, 1997] von Randow, G. (1997). Roboter. Rowohlt, Reinbek.

[Walter, 1950] Walter, W. G. (1950). An imitation of life. Scientific American, 182: 43-45.

[Walter, 1951] Walter, W. G. (1951). A machine that learns. Scientific American, 51: 60-63.

[Wilcoxon, 1947] Wilcoxon, F. (1947). Probability tables for individual comparisons by ranking methods. Biometrics, 3: 119-122.

[Wiltschko and Wiltschko, 2003] Wiltschko, R. and Wiltschko, W. (2003). Avian navigation: from histgorical to modern concepts. Animal Behaviour, 65: 257-272.

[Wolf, 2003] Wolf, A. (2003). Chaos analysis software. http://www.cooper.edu/_wolf/-chaos/chaos.htm.

[Wolf et al, 1995] Wolf, A., Swift, J., Swinney, H., and Vastano, J. (1995). Determing lyapunov exponents from a time series. Physica 16D.

[Wyatt and Demiris, 2000] Wyatt, J. and Demiris, J., (Eds.) (2000). Advances in Robot Learninng: 8th European Workshop on Learning Robots. Springer-Verlag.

国际视野　科技前沿

☆ 原书由自主移动机器人领域知名人士Ulrich Nehmzow博士撰写，在英国很受欢迎，并且在意大利、西班牙和加拿大等国被多所高等院校作为教材使用。

☆ 本书基于动力系统理论、混沌理论、神经网络、统计学和系统辨识等理论，第一次运用数量分析和计算机模型对机器人、任务和环境三者互动的行为建立了完整统一的公式化理论研究体系。这是一个全新的前沿视角，也是解决此类长期困扰我们的复杂问题的全新途径。

☆ 作为反映最新前沿技术的书籍，全书从基本概念到应用实例进行了系统的阐述，内容全面、重点突出、理论与实例并重，叙述风格深入浅出，能够很好地将理论介绍给相关领域的一般研究人员和专家学者。

☆ 本书可供从事机器人相关工作的科研及技术人员参考，也可作为高等院校机器人、自动控制、计算机、人工智能、模式识别和生物医学工程等专业的机器人课程或人机交互行为课程的教材。

上架指导：工业技术／机电一体



地址：北京市百万庄大街22号
电话服务
社服务中心：(010)88361066
销售一部：(010)68326294
销售二部：(010)88379649
读者服务部：(010)68993821

邮政编码：100037
网络服务
门户网：http://www.cmpbook.com
教材网：http://www.cmpedu.com
封面无防伪标均为盗版

ISBN 978-7-111-28976-0
封面设计：鞠杨
定价：38.00元